SAPIENZA
UNIVERSITÀ DI ROMA

NetGrowth: simulating the growth of biological neurons in spatial confinement

Facoltà di Scienze Matematiche, Fisiche e Naturali

Corso di Laurea Magistrale in Fisica

Candidate

Alessio Quaresima
ID number 1471802

Thesis Advisors

Prof. Samuel Bottani
Prof. Paolo Del Giudice

Academic Year 2017/2018

Thesis defended on 17 April 2018
in front of a Board of Examiners composed by:

Prof. Egidio Longo (chairman)
Prof. Giancarlo Ruocco
Prof, Paolo Calvani
Prof. Francesco De Luca
Prof.ssa Valeria Ferrari
Prof. Franco Meddi
Prof. Stefano Sarti
Prof. Paolo Pani

NetGrowth: simulating the growth of biological neurons in spatial confinement
Master thesis. Sapienza – University of Rome

This thesis has been typeset by LaTeX and the Sapthesis class.

Version: April 12, 2018

Author's email: q.ale@protonmail.ch

Á les souris de Paris

Abstract

Advances in biological comprehension and technology allows for more complex in vitro experiments. C. Villard's group at IPGG in Paris [Renault, 2015] as well as other authors [Smirnov, 2014] performed studies on the environmental interactions between growth cones and different substrates and mechanical confinements. The results of such studies are as relevant for medical applications as for the fundamental understanding of growth cone inner mechanisms. Furthermore, the investigation of the neuronal code brings researchers to study spontaneous activity in vitro, shedding light on the bursting phenomenon and it's spatial properties [Orlandi et al., 2013a]. To reproduce such dynamics, a useful tool as NEST can be used – a powerful open-source simulator reproducing neuronal activity with eligible complexity. However, no tool is currently available to reconstruct the network morphology or topology with a biological hint.

The Neurophysics group of Université Paris VII-Diderot collected theories and results from biophysics and biology in order to combine them into an efficient neuronal growth simulator: NetGrowth. This simulator is built up with the idea of modularity, allowing the researcher to choose the complexity level of his simulation. Since the exact dynamics of actin, tubulin, and other constituents is still obscure, the simulator can act as a valid instrument to perform in silico experiments.

The application of such a tool is broad, ranging from the microbiological laboratory, looking for a benchmark for in vitro experiment, to theoretical physics, requiring more realistic topology and morphology to study in silico dynamics.

At this stage we tested some models for environmental interaction and elongation mechanisms, starting from biophysical theories and biological evidences. The navigation model results from an advanced study of random walk and run-and-tumble algorithms. For the elongation we implemented and studied the dynamics of a double SDE system, which reproduces the perishable resources the growth cone requires to stretch its skeleton and to elongate itself. The branching mechanism is simulated as well. We have implemented state of the art models in an efficient manner. The relation among diameter and branching angles is reproduced as observed in the laboratory. The environment sensing model – the most relevant novelty of the project – is implemented as a functional convolution of probability. The interaction between neurons which leads to fasciculation is still a work in progress. Eventually, the NetGrowth simulator is able to produce network topologies. The actual method for synapse establishment is a naïve estimation of neurite density, which was already tested for other environment-agnostic simulators.

The manuscript describes in details the state of the art, the study behind the realization of the simulator and the obtained results.

Acknowledgments

Eh, scrivere i ringraziamenti per un lavoro così intenso che ha segnato tanta parte del mio ultimo anno, e che segnerà, in qualche modo, il mio futuro a venire, è poco meno che scrivere i ringraziamenti alle persone importanti della mia vita. Quindi procederò in ordine sparso, così non si offende nessuno.

Innanzitutto voglio ringraziare Mila: una volta mi rivelò di essere curiosa di sapere cosa pensasse il suo cavallo. Per soddisfare quella semplice curiosità mi iscrissi al corso di neuroscienze teoriche tenuto dal professor Del Giudice. Non seppi risolvere il suo enigma, ma in compenso, mi innamorai delle neuroscienze. Dinanzi allo spettacolo del cervello tutto il resto scoloriva, a cosa serviva la fisica se non per indagare le infinite possibilità e fragilità della materia pensante? Ringrazio il professor Del Giudice per le sue lezioni, per l'oratoria, e per aver con pazienza e comprensione seguito la scrittura di questa tesi. Uno speciale ringraziamento anche a Samuel e Tanguy che mi hanno seguito con meticolosa apprensione in questi mesi, sono stati la prima esperienza di vera ricerca, spero che possa sempre trovare tanta umanità, onestà e amicizia. Grazie

Eppoi tutti gli altri. L'Officina di Fisica, perché ha segnato il mio percorso in questo dipartimento e soprattutto mi ha convinto che 'a na certa è meglio finire. Giulia per la pazienza, l'affetto e il sorriso. I miei genitori, per la pazienza, l'affetto e i soldi. Alessandro e Davide, per le piacevoli chiacchierate e le dritte. Simone, Giacomo, Elisa, i saltatempisti e tutta la provincia di Roma che sostiene le mie radici e offre un sicuro porto e ristoro, vi voglio bene. Blu ché mi ha seguito in questo percorso con insidiosi dubbi. L'Officina di Fisica ancora, perché oltre ad essere compagni di sventure sono amici e amiche che mi hanno sostenuto con incoraggiamenti, consigli e sguardi solidali. AvANa, Hackmeeting, l'InsomniaLab e tutti gli acari, tutti quelli che pensano che la tecnologia non è neutrale, ché manco la scienza è neutrale, questo l'ho capito bene. The Science Zone e tutti ile tszonie, fonte inesauribile di amore per la scienza. Claudia, che mi ospita con affetto ed è la scienziata più scienziata che conosco. Pablo qui m'a toujours demandé -comment sont le neurons?; alla Senna, ai ragazzi del MSC Lab, a Pascal e Stephane, agli amici e le amiche a Parigi, al Saint Savoir di Rue Panoyaux. Un sentito ringraziamento alla mia famiglia, a Martina, a Luca e Elisa, che possano trovare nel mio lavoro ispirazione e curiosità; per quanto abbia poco a cuore le faccende di sangue, voi ci siete sempre e mi date la serenità necessaria a fare qualsiasi progetto di vita. Un saluto affettuoso anche a Giulia e Manuel e Francesca, un saluto a Trieste. Eh, chissà se poi lo vinco il dottorato... Un ringraziamento affettuoso a Karla, Asya, Carlos e ai ragazzi del NeuroAspect di Varsavia, che hanno accettato il mio poster e io mi credevo chissà ché e invece son finito in un corridoio affianco ad altri cento disperati come me che spendono la loro vita dentro un laboratorio, davanti al computer, sopra ai libri, e tutto per fare un millimetro in più sulla Grande Autostrada della Conoscenza. Che poi dove porti st'autostrada non s'è capito, ché magari bastava uscire al KM 42 e potevamo star tutti in spiaggia a giocare a Beach Volley. Grazie a tutt eh!

Alessio

Contents

Some notes on my internship at MSC Lab

Before get into the work itself, I want to submit some considerations: When I started this project one year ago I was an Erasmus student, with a curriculum in theoretical physics. My experience wasn't net neither, indeed my Master Degree has been much more a survey over whole Physics than a proper specialization course. I get involved in this project at NeuroPhysics group without a clear comprehension of its purpose, I liked the people, and that was enough. When I arrived in Paris, I needed a full month to understand what I was expected to do, and when I realized I was scared by the huge amount of work. I was used to program in C++, not to build user-proof interfaces. The aspect that baffled me most was the biology of growing neurons. All the knowledge in biology I had dated back to high school and the course of Theoretical Neuroscience by prof. DelGiudice. I remember those days, exactly one year ago. When I expressed my doubts to Samuel, my mentor, he listened to my preoccupations and encouraged me, he said that it was not as hard as it looked, that, step by step, I would build the simulator and be surprised by the work and difficulties I had overcome. And so it was. It was hard, sure, I had to study an open problem like neuronal growth, study it through an incredibly vast literature and a wide set of point of views, discovering reviews and articles day by day, dropped off the clouds when a new report contradicted the one I was trusting. Yeah, it was hard. But I never felt the fever of science as in those days spent with Samuel and Tanguy discussing on the future implementation of the simulator. I have never learned so much about Science than during the days reading neuroscience review. I am thankful to Samuel, Pascal, Tanguy and Stephane for the confidence they gave me.

Part I

Neuronal cultures as computational units

Here we begin this journey in the maze of neuronal computation. What is *computation*? Is it some special feature of Intel produced chip-set or a basic property of matter? What happens in our brain when we sum up the costs of a bill? Is it related to the recognition process of a friend's face? Where does the computation happen in our brain?

With this project we wanted to focus on a very simple problem: is it possible to reproduce such complex operations inside a 2D culture? If so, which are the required ingredients? Is it the network topology, its geometry, or the strength of the synaptic connections? And how does varying these elements affect the properties of the system? These questions are on the edge among epistemology, artificial intelligence, and neuroscience. The quest is on the neuronal code i.e. the actual method used by neurons to communicate with each other, which is a top-shelf question in neuroscience. Let's start from the basic ingredients to pave the way for the NetGrowth simulator.

Manuscript outline The thesis deals with the design and realization of NetGrowth, a neuronal growth simulator based on biological evidence. The project is maintained by the NeuroPhysics group, at the MSC Lab, Paris Diderot University. The work is divided in three parts: The first part is an introduction to the project itself, the state of the art is described and the research project is motivated. The first chapter is an overview of biologically based computation, it is a survey over the research project of the NeuroPhysics group. The second chapter focuses on *in vitro* neuronal cultures, the results obtained with different experimental techniques, some attempts of reproducing computational units with culture are illustrated. The third chapter presents some pre-existent simulators and list their pros and cons, the relevance of NetGrowth in this panorama is discussed.
The second part presents the work we have done in the last year, during the internship and in subsequent months, to develop an efficient simulator. The first chapter gets into details of neuronal microbiology, illustrates elements and processes we wanted to reproduce with NetGrowth. The second chapter reports some biophysical models we have based our work on. Eventually the third chapter is the richest and the most demanding of the whole work, this chapter illustrates and evaluate the model used for the simulator, the most of original work I have done during the internship is collected in this chapter. In the fourth chapter NetGrowth is evaluated and discussed, some complex scenarios are presented and future improvements are outlined. In the conclusion I evaluate the relevance of the project and possible outcomes.

Chapter 1

Computation as an emergent property of networks

1.1 Neural or Neuronal Networks

The very first concept to deal with is *information*. When Shannon introduced the information theory in 1950 [Shannon, 1948] he couldn't imagine it would have affected so much our contemporary culture. Indeed information is nowadays one of the most profitable and abused concept. Information is the amount of knowledge we have on a certain system, the density measure of information, as introduced by Shannon, is its entropy. The theory was formulated in the field of Communication Theory and aimed to formalize the transmission of discrete signals; in this particular case, the entropy describe the uncertainty about the next character of the transmitted string. The concept of information was soon related to *computation*: the ability of process such information, to extract relevant parts from the whole, detect similarities and make evaluations. Living beings exert these actions all along their life.

The general quest about how the living matters process the information is on going debate, the discussion is even pushed further to comprehend the nature of consciousness. Anyway a meeting point between Neuroscience and Informatics are neural networks, an oriented graph of processing units. Neural Networks are driving the Informatics research on Machine Learning and obtaining the most striking results in reproducing *intelligent* behaviour[1], attracting the attention of researchers and stakeholders.

The common points between a formal system, i.e. ANN, running on top of Silicon hardware and the soft matter in our brains are the network structure and the existence of an activation threshold. A simple sketch of this parallel is offered in Fig 1.1. One of the most important aspects of this analogy is that *Computation* appears not as an intrinsic property but emerges from the web of connections among single, simple, units.

Computation, as every one knows, is the fundamental feature of computers, which are composed by transistors, and in the very end, by a well-structured amount

[1] The most famous was AlphaGo winning the world champion of Go, in 2016.

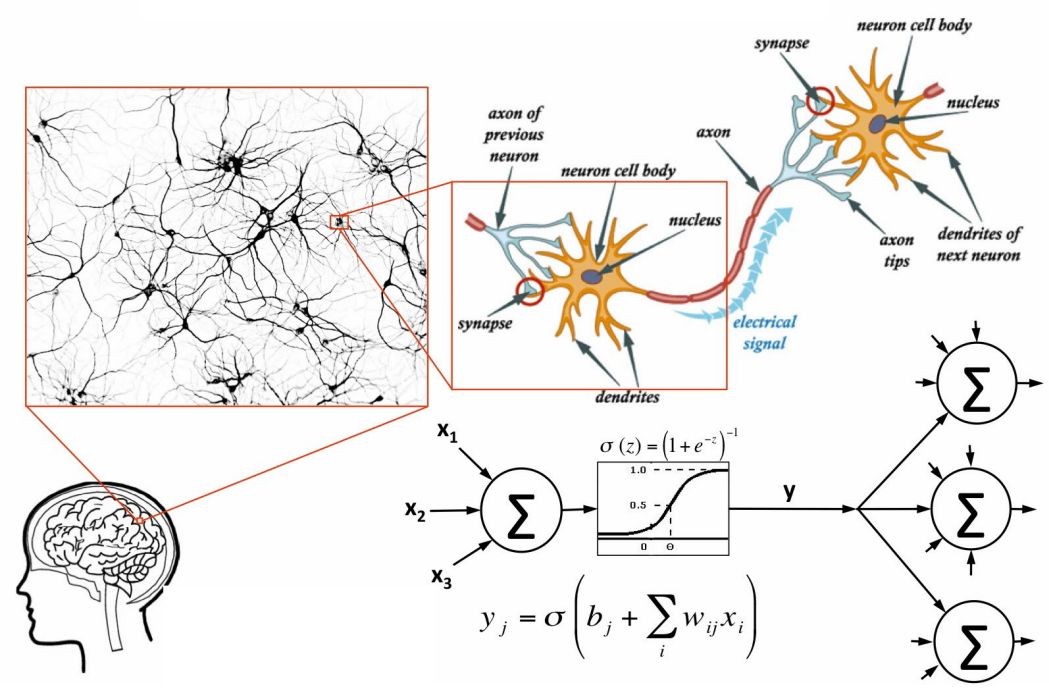


Figure 1.1. Neural or Neuronal Networks The parallel between Artificial Neural Networks and biological NN can be sustained on two main evidences. First both the networks are built upon and with simple elements whose computational properties are minimum, they can usually sum a set of input and confront with a threshold. The inputs can be both excitatory or inhibitory. Second, a common feature is the non linear response. The all or none behaviour is defined by an activation function $\phi(x)$, which is a sigmoid for the artificial networks and variate by neuron to neuron in biological tissue. On the other end a relevant difference is the time domain the agents act on. For neuronal networks the information appear to be coded in time also, while the ANN uses discrete, where each evaluation takes one step. *Thanks to McCulloch & Pitts youtube channel*

of Silicon. The way these systems process the information is well known, studied and improved day by day into researches lab. Processors, complex structures of hardly more than nanometres components, are among the most advanced product of human technology, they are composed basically by an ALU (Arithmetic Logic Unit) and a CU (Control Unit) the former operates those actions we usually regard as computation while the latter control the first, shift information and manage the whole process.
From a mathematical point of view, the work of Alan Turing (1912- 1954) laid the foundations of computer science, and fixed the constraints to what machines can do or not[2]. In 1980's the introduction of quantum computer changed profoundly the panorama, and new categories and tasks were defined, changing completely the panorama of complexity classes, and redefining the concept of *hardness*. Maybe a better comprehension of biologically based information process will help to solve some of the most puzzling questions of our century and shade the light on the many doubts of Infromatics.

1.2 Graphs in Neuroscience

The relevance of graph structures appear in various field of Neuroscience. The network framework provides a natural way to describe neural organization. Indeed, information processing emerges from the activity of neural networks that carry information from one cell assembly or brain region to another. The advent of Network Science suggests modifying the traditional *computer metaphor* for the brain to an *Internet metaphor*, where the neocortex takes on the task of *packet switching* [Baronchelli et al., 2013]. Network theory allows the shift from a reductionist to a *complex system* view of brain organization. In this framework, optimal brain functioning requires a balance between local processing and global integration [3]. I will shortly present some examples:

Convergent or Divergent Connections The very interesting work of Martin Lindauer [Lindauer, 1996] starts from networks and synapses to reveal the complex simplicity of animals communication. In the account of sensory systems Lindauer depicts the difference between divergent or convergent networks on the sensor nerve. When the signal from a cell is shared with upper standing cells the information is reinforced and the possibility it arrives to higher areas of the brain increases. It can also be processed and refined, as in the interneurons of eye's nervous system which convolve many signals from the retina to obtain the *contrasted* image. The convergent configuration, instead, allows for more complex information: the perception is augmented mixing different signals, e.g. the bees antenna mixes olfactory and

[2]Turing introduced the notion of Turing complete machines, and showed the same, codified, computation can be performed on different architectures changing the particular set of procedures, the algorithms. Together with Gödel they set the limit of such machines by the very famous Halting Problem: for a undefined set of inputs a Turing machine can not say whether it's computation is going the end or not, these problems are said undecidable.

[3]`http://www.scholarpedia.org/article/Integrated_information_theory`

tactile signals for a plastic perception of flavours. Fig 1.2 present a schema of the connections types.

Connectomics This field focuses on the production and study of the *Connectomes* [4]. These are comprehensive maps of connections within an organism's nervous system, typically its brain or eye. The human connectome can be viewed as a graph. Therefore, graph theory by its rich tools, definitions, and algorithms, is commonly used. By comparing diseased connectomes and healthy ones, it is possible to get insight into certain psychopathologies - such as neuropathic pain - and potential therapies. It is nowadays commonly accepted that the cognitive abilities and the information processing of brain's tissue resides into the variate and intense connection among brain areas. The connectome is characterized by short path lengths (a small-word topology), high clustering, and assortativity, the tendency of hubs to be connected to hubs, forming a so-called *rich club*, and an overlapping community structure [Baronchelli et al., 2013].

Integrated Information Theory When talking about information processing and biological intrinsic properties, IIT plays the most distinguished role in the attempt to formulate a comprehensive theory of human consciousness. IIT lies on graph theory and precisely on the convergence between graph theory and Information theory. The topology of the network is the most important aspect. Roughly speaking, the possible partitions represent the possible states of the experience [Massimini, 2013].

1.3 Morphology and computing dendrites

The importance of graph topology comes in action when moving to the single unit too. We have briefly mentioned the simplicity of the fundamental component of neuronal networks i.e. the nervous cell. However, the main attention of the work in NetGrowth regards the complex topology and geometry of dendritic trees. The reader who is unfamiliar with neuron biology will find a short review in Fig. 1.3. Hence, in the following paragraphs I will present some examples of network. Such systems due their properties to the topology and morphology of networks they are composed of.

A hint of the complex process of arborization in offered in the comprehensive review in [Cuntz, 2014]. The dendrite is investigated as an active actor of the neuron-neuron signaling process, and its properties are outlined. This work was a good basis over which to build our simulator, and it is eventually to test the result we are obtaining. The chapters we were most interested are those regarding the optimization principles the neurite satisfy during the growth. The physical quantities which constraint the neurite outgrowth are linked with its biological limits and scope: the maximization of neuron connectivity given its limited set of time and nutrients [Cuntz et al., 2010],[Tsigankov and Koulakov, 2009], e.g. the maximization of synaptic repertoire given the dendritic arbor total length.

[4] `http://www.scholarpedia.org/article/Connectome`

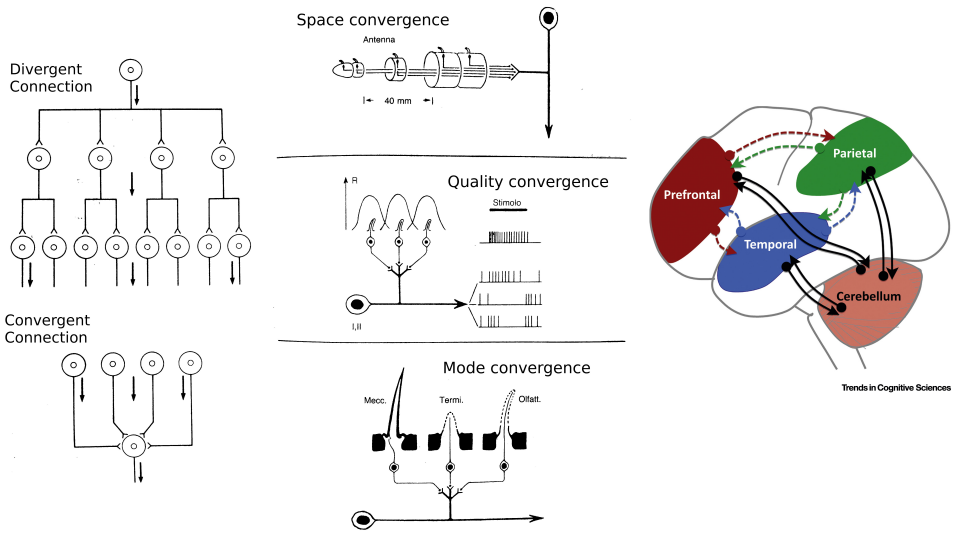


Figure 1.2. Network-Theory approach in Neuroscience Network Theory offers an effective framework for the study of information processing properties of neuronal tissue. Mapping the connections among populations, or even among cells, into a schematic graph helps understanding some foundamental properties, as heterogeneity of connections, repeated patterns, node-node distance, etc.... The pictures reproduce experimentally observed networks in two different scale of neuronal tissue.
Left picture: scheme of nervous connection Lindauer observed in nervous systems of insects and mammalians. Sensor cells have two type of connections, usually both present in same sensory unit. The divergent connection ensure redundancy and signal transmission, while the convergent connections gives rise to complex stimulus when merged into an interneuron. On the right some examples of converging connections, each one with a specific function [Lindauer, 1996].
Right picture: The cerebellum receives input from cortical areas via the pons and projects back to similar areas via the thalamus, forming a closed-loop architecture (black arrows). Complex cognitive processes such as language or social cognition require interactions between distributed regions in the cerebral cortex (colored broken arrows).[Sokolov et al., 2017]

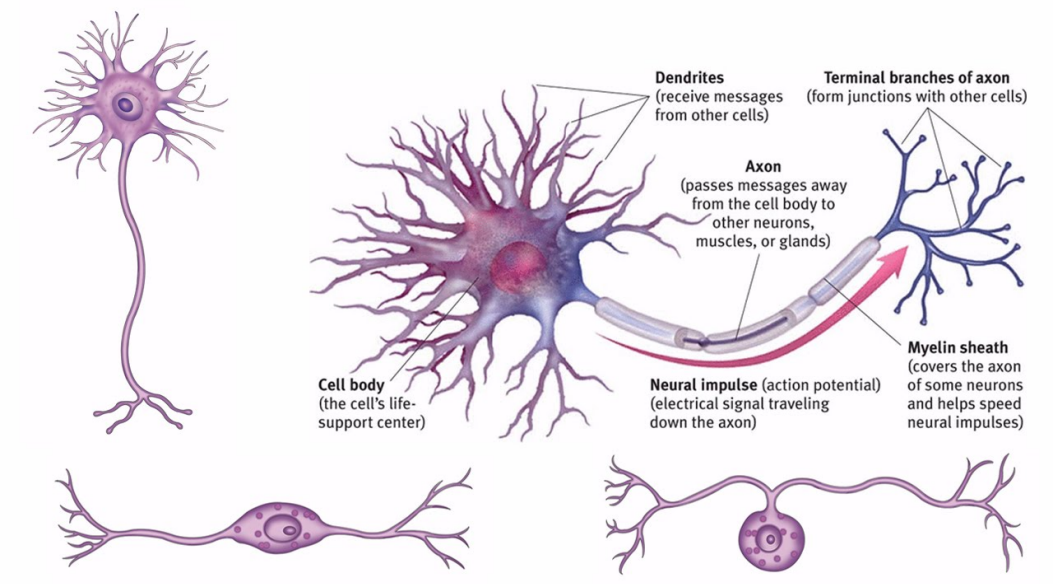


Figure 1.3. Scheme of neuronal structure The picture presents a schematic neuron with all its fundamental components. The neurons communicate with each other throughout a electrochemical signal which is originated in the soma (cell-body) and travels across the axon up to the neurite of the post-synaptic neuron. The dendrites transport the signal towards the soma and play a significant role in amplifying, attenuating and preprocessing the signal. The synapses are the connection spots between axon and dendrites. About a million of synaptic connections can arrange over a mature dendritic tree.

The properties of the dendritic tissue are relevant for signal transmission and processing also; Nonetheless the early theories about dendritic tree, i.e. only a signal collector, it was suspected its role to be significantly richer, in light of the very typical dendritic architectures of some neuronal types. The presence of voltage-gated ion channels and action-potentials inside the dendrites, nowadays observed, obliterates the idea that they merely funnel incoming signals passively to the soma. Interestingly, it was shown that – under particular conditions – dendritic spikes can compensate for the distance-dependent attenuation observed in passive dendrites, so that the efficacy of synapses is nearly location-independent [Velte and Masland., 1999]. The function of active dendrites could therefore be a simple way for neurons to free themselves from the strong metric constraints inherent to physical networks.
But a neuron endowed with excitable dendrites could do much more and perform a complex non-linear summation of its synaptic inputs, depending on their spatial and temporal activation pattern. Although the importance of active dendritic trees has been predicted for a long time, their role in computation has been experimentally observed quite recently. In the retina, for instance, direction selectivity was shown to depend on active dendritic processing. Theoretical investigations showed that – under certain circumstances – even neurons with passive, non excitable dendrites can compute non trivial functions (like the XOR logic gate), so that dendritic (pre)processing might be a general feature throughout the nervous system rather than an exception.

In practice it might not be evident to exploit such computing power in artificial devices because it would presumably require precise positioning of synapses on the dendritic tree of each neuron, which is firstly technologically challenging and secondly intrinsically regulated through non yet elucidated mechanisms. The description of this problem lies outside the scope of the manuscript but underlies the importance of a plausible reproduction of neurite morphology. A short resume on well-known processing abilities is reported in Fig. 1.4.

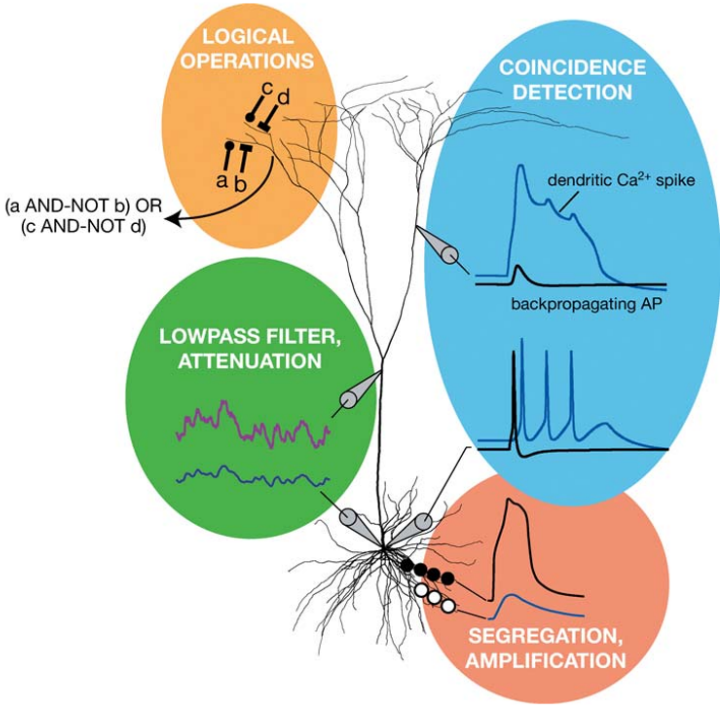


Figure 1.4. The dendritic computational toolkit A schematic figure highlighting four key dendritic mechanisms, mapped onto a layer 5 pyramidal neuron morphology, which can allow dendrites to act as computational elements. These mechanisms can coexist in the same neuron and be active in parallel or in a hierarchical manner.
Bottom left: Passive dendrites act as passive filters. A high-frequency fluctuating current injected in the dendritic pipette will evoke high-frequency and large-amplitude local voltage responses, but the response recorded by the somatic pipette will be attenuated and smoothed (low pass filtered).
Top left: Nonlinear interaction between excitation and shunting inhibition on small dendritic branches can implement logical operations. The branch marked by an arrow sums up the current from the two subtrees, such that its output would be a logical OR on their output. Each of the subtrees will in turn inject current if and only if the excitation AND-NOT the inhibition will be active.
Bottom right: Dendrites can help reduce or amplify the mutual interaction between excitatory inputs. Excitatory inputs to the same branch tend to sum sublinearly, whereas inputs on different branches sum linearly. Thus mapping input to different branches can reduce this effect. In neurons with active dendrites, however, clusters of inputs active synchronously on the same branch can evoke a local dendritic spike, which leads to significant amplification of the input. Synapses onto a different branch (open circles) are only slightly influenced by this spike.
Top right: In layer 5 cortical pyramidal neurons, as depicted here, coincidence detection between the apical and basal dendritic compartments is achieved by active dendritic mechanisms. A backpropagating action potential, which coincides with a distal synaptic input, will trigger a dendritic Ca 2+ spike, which depolarizes the whole apical dendrite and drives a burst of spikes in the axon.[London and Häusser, 2005]

Chapter 2

What are neuronal cultures and devices

In this chapter we will move from the abstract description of network and computation in brain tissue towards an effective investigation of computational abilities of cultured neurons. We are interested in understanding the basic, minimal, properties of neuronal tissue, and we will concentrate our research on *in vitro* systems. In the following we will describe briefly what are neuronal cultures and how they can be arranged to form a functional circuit.

Before to start, some remarks. Leaving well-behaved models and semi-conductors for the realm of biology is easier said than done. Using living materials imposes a lot of constraints to the experimenter who wishes to build something out of them. As everything that is alive can die, the first challenge is to keep these living materials healthy as long as necessary. Providing a physiological environment with the appropriate nutrients is of course the main determinant for survival. Like they do in vivo, cultured neurons are subjected to aging, giving as a consequence an additional layer of temporal constraints. The development and maturation of cultured neuronal networks actually involves a large panel of processes through which neurons first grow and establish synapses with other neurons, then express different sets of genes as time goes, causing some major shifts in their electrophysiological properties. Provided that the basic needs of neurons are satisfied, and that they are in the right developmental stage, there is another hurdle: homeostasis. Cells that we grow in vitro have originally evolved different compensatory mechanisms to maintain an internal state that guarantees proper operation across a wide range of environmental conditions, for the sake of their common good, that is the survival of the animals they normally constitute. The cells will continue to fiercely regulate themselves when grown in a Petri dish, whether or not the functional implications are relevant to the experimenter. The sad corollary is that properties of interest that are incidentally connected to these homeostatic mechanisms can hardly be controlled experimentally, at least under physiological conditions. The usage of microfludic chamber soothe this issue.

A closer look to this experimental setup is offered in the next section.

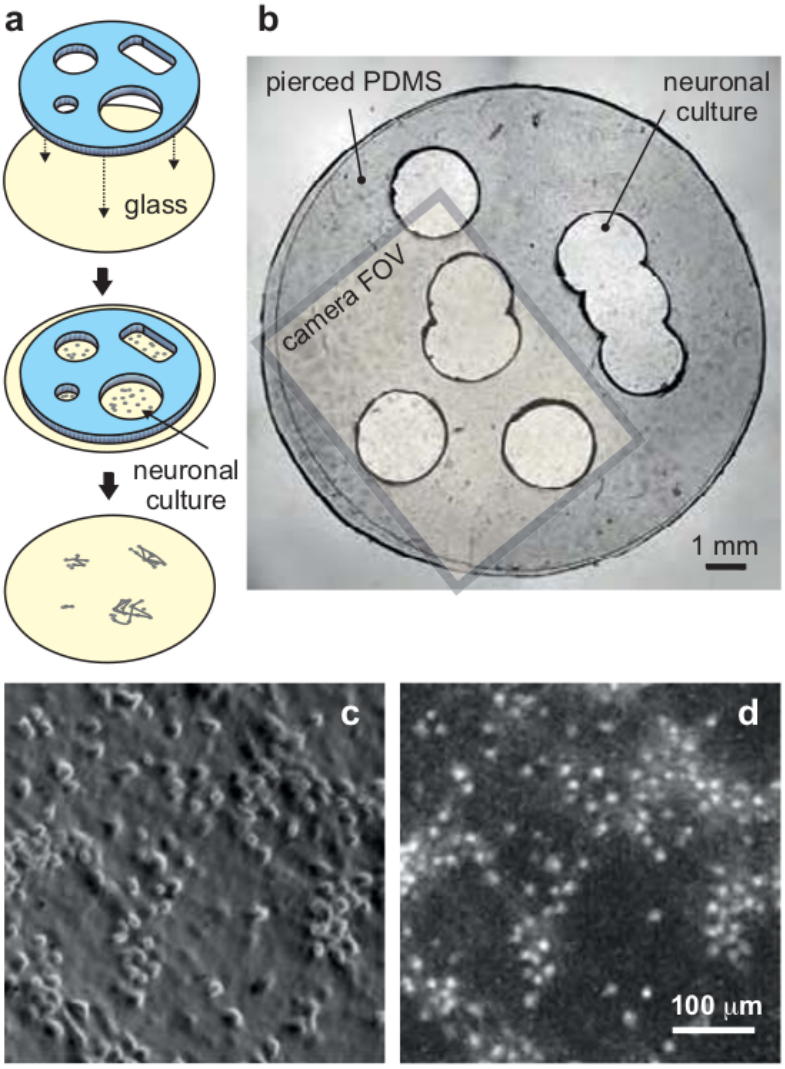


Figure 2.1. Neuronal cultures Neuronal cultures are an excellent tool to investigate arising neuronal networks and activity dynamics. Various types of measures can be obtained from the culture, from single cell activity up to a global picture of networks dynamics with Ca++ Imaging. The modern techs in soft lithography permit finely modeled structures, and microfluidic are used to feed the cells. Nevertheless the observation of activity *in vitro* cannot be directly compared to *in vivo* studies because the lack of sensory stimulus and the different homeostasis affect irreversibly the population activity. **a** Schematic representation of the culturing process by using pierced PDMS molds (blue) attached to glass coverslips (yellow). The preparation process (top) included piercing, autoclaving and polylysine coating; the culturing process (center) yielded to the formation of mini–cultures in the pierced areas; the final process (bottom) consisted in the removal of the PDMS mold and the preparation of the culture for the measurements. **b** Actual image of combined glass–PDMS structure 11 days after plating. The rectangular area depicts the maximum field–of–view (FOV) of the camera (8.2 x 6.1 mm) used in the experiment.
c Bright field image of a small region of a culture. Round objects are neurons' cell bodies. [Orlandi et al., 2013a]

2.1 Basic description of cultures: fabrication, properties, applications

A cultured neuronal network is a cell culture of neurons. It is used as a model to study the central nervous system or to investigate fundamentals properties of the living matter. A scheme is depicted in Fig. 2.1. The fabrication stages are generally based on soft litography, which enables for resolution at tens of nanometers scale, the most common material is the PDMS (Polydimethylsiloxane), an elastomeric organosilicon with notable visco-elastic properties. Soft litography has a wide set of advantages over the traditional litography techniques and is widely used in biotechnology. The properties of PDMS include the possibility to build sharp edge structures where neuron can be confined. Neuronal networks are typically cultured from dissociated rat neurons, because of their wide availability. Studies commonly employ rat cortical, hippocampal, and spinal neurons although laboratory mouse neurons have also been used. One of the most formidable problems associated with cultured neuronal networks is their lack of longevity. Like most cell cultures, neuron cultures are highly susceptible to infection. The long timelines associated with studying neuronal plasticity – usually on the scale of months – make the extension of the lifespan of neurons *in vitro* paramount. The applications of such devices range from medical to fundamental research. Starting from the laboratories which I personally visited or acknowledged, they include basic studies on axon guidance and interaction with chemical and mechanical environment [1], graphene based neuronal tissue reconstruction or 3D growth from PDMS micro-pored structures [2], and full network studies as those performed on bursting spatial patterns [3]. The flaws of neuronal cultures are numerous, not even counting the difference in terms of lifespan timescales. Cultured neuronal networks are by definition disembodied. Therefore, the neurons are influenced in ways that are not biologically normal when outside their natural environment. Among these abnormalities is the foremost fact that the neurons are usually harvested as neural stem cells from a foetus and are therefore disrupted at a critical stage in network development. Lacking a body, neurons are also devoid of sensory inputs as well of the ability to express behaviour – a crucial characteristic in learning and memory experiments. It is believed that such sensory deprivation has adverse effects on the development of these cultures and may result in abnormal patterns of behavior throughout the network. Traditional cultured networks, 3D structure is a promising exception, are flat, single-layer sheets of cells with connectivity only two dimensions. Most in vivo neuronal systems, to the contrary, are large three-dimensional structures with much greater interconnectivity. This remains one of the most striking differences between the model and the reality, and this fact probably plays a large role in skewing some of the conclusions derived from experiments based on this model.
The spectrum of accessible experiments and application has increased with the introduction of more sophisticated techniques of production. The Microfluidics has emerged as an important technology in academic research and even more remarkably

[1] Villard, Ipgg lab, Paris [Renault et al., 2016]
[2] Ballerini, Sissa, Trieste [Bosi et al., 2015, Fabbro et al., 2012]
[3] Soriano, UBV, Barcelona [Orlandi et al., 2013a]

in the industry, to control the architecture of neural networks growth *in vitro*. Using microfluidic devices offers indeed many advantages over simple patterning techniques, including high manufacturability, robustness, controllable environment, etc...
In the next section a basic idea of tools and methods used to study neuronal culture is presented, then some highlights on the activity of the neuronal culture are reported and eventually the section is concluded with some notes on a particular type of culture: the one produce at IPGG and whom I will refer often throughout the manuscript.

2.2 Instruments to observe and study neuronal cultures

The investigation of neuronal cultures is accomplished with several techniques, whose perspectives and invasiveness change appreciably. I had the chance to study and, in some case, perform some experiments during the Neuron Technology Summer School in Trieste, two weeks of seminars and hands-on sessions focused on the experimental techniques of neurobiology.

Single neuron activity investigation can be conducted in the field of electrophysiology: measuring the electrical potential at various position in the cell's shaft uncovers the main communication pathway for neuronal networks. Electrophysiology can access the voltage profile of a single neuron, or larger areas field potentials. These experimental methods are the most long-lived in experimental neurobiology and where fundamental for the formulation of neuronal dynamics models. The intrusiveness of such experiments depends on the setup, but usually the single cell recordings lead the cell to death in the range of a few hours.
A wider perspective can be obtained with the Multi Electrode Arrays (MEA), at the cost of less resolution. MEA are grid structures of in/out electrodes. They allow for largely attenuated and temporally filtered signals as well as simultaneous recordings of large populations without cell labelling. The electrode diameter is 100 μm and a standard component is Silver Chloride, which converts ionic current into electricity. Reversely, it is possible to electrically stimulate neurons through the electrodes. MEA is not recording the internal properties but the local field potential (LFP). The LFP signal is blind to the subtreshold area of the neuron activity, and EPSP or IPSP-released present the same LFPs . Recent approaches consist in changing the shape of the array because the plate shape has a limited coupling efficiency. When the morphology, the topology, or the overall network dynamics is the investigated property the electrophysiology gives way to imaging techniques. The morphology, for instance, can be investigated by fluorescent microscopy, e.g. Audric [4] used *SiR* fluorescent dymers to evaluate tubulin amounts in outgrowing neurons. An exceptional investigation tool has recently been introduced with Calcium Imaging [Grienberger and Konnerth, 2012]. The Ca++ intracellular current occurring in spiking neurons proxies for a spatial localized pictures of activity. The Calcium Imaging is widely used to test for pharmacological response, for bursting patterns, and for numerous other experiments that associate single neuron labelling (i.e. by its position in the culture) to culture dynamics on very large scales. Calcium Imaging can be combined with MEA for high quality spatial profiling of culture electrical

[4] In the context of his Report des stage, which is not available online

response. There are severe flaws for this method too. First, the Ca++ marking dimer decays over time and poisons the cell. Second, the measured signal is coarse in time and is attenuated on scales much longer than the neurons' spikes. The Calcium signal cannot eventually distinguish a inhibitory signal from an excitatory one.

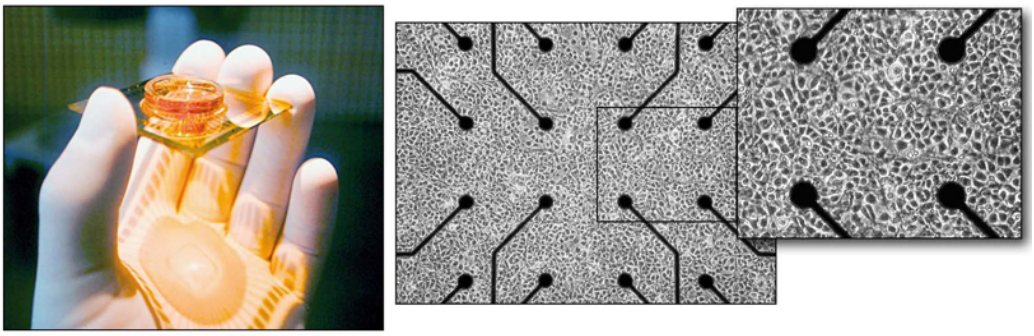


Figure 2.2. MEA culture dish and close-up look on the electrodes. The electrostimulation of cultured neurons can be arranged as a learning protocol for the network. LTSP and LTSD (long term synaptic potentiation or depotentiation) can be triggered with external stimuli from the electrical grid placed at the bottom surface of the culture. These arrays are generally composed by tens of micrometers electrodes which can stimulate the tissue and track the response.
The picture shows a weekold culture of circa 50000 neurons and glia from embryonic rat cortex, growing in a MEA and forming a dense network 1-2 mm across. Fifty-nine 30 μm electrodes spaced at 200 μm intervals connect a few hundred of the network's neurons to the outside world, by allowing their activity to be extracellularly recorded or evoked by electrical stimulation.

2.3 Activity in neuronal cultures

When the culture is constructed and the neurons displaced on the Petri dish, the complex process of growth takes place. This process itself is detailed in the thesis and the scope of NetGrowth is to reproduce it.
The importance of electrical activity in the shaping of the network topology is well assessed in the neurobiologist community, e.g. neurons die precociously when failing to connect, but this topic belongs to the future perspective of this research project and will not be tackle. Here, we want to discuss the electrical activity that animates the mature culture, once the growth is coming to an end and before their feeble life ceases. The literature on this topic is wide, for simplicity I will concentrate on the experiments the group of J. Soriano is pursuing in his lab in Barcelona.
Once the most of synapses are established, and the topology becomes more constant in time, the culture enters in the bursting phase. This phase is essentially dominated by the spreading of activation waves throughout the culture cell; these waves can be observed, e.g, with Calcium Imaging. The generation of these waves is investigated both theoretically [Fardet et al., 2018] and experimentally [Orlandi et al., 2013a] and reveal many similarities with the physical phenomenon of coupled oscillators: a sufficient condition for network bursting is the presence of an excitatory population of

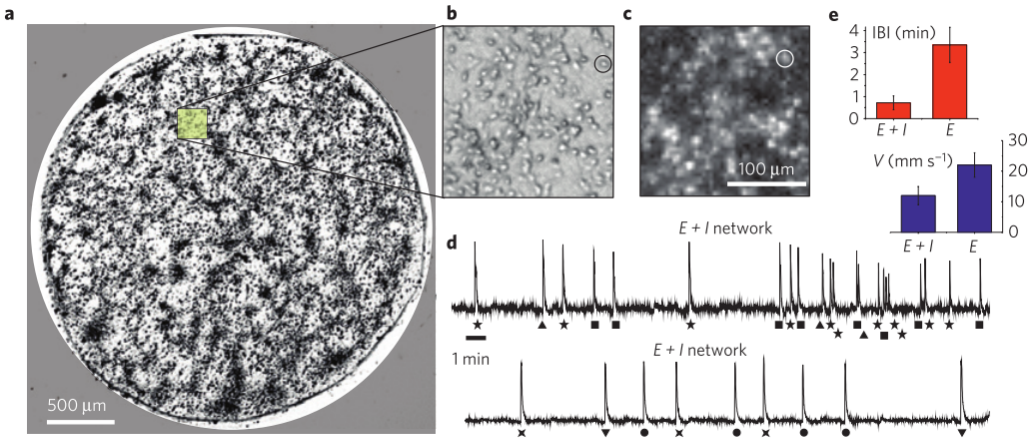


Figure 2.3. High-contrast bright-field image of a neuronal culture in Ca++ Calcium Imaging permits to measure the spatial localization of tissue activity. Nonetheless the resolution is not at spikes' timescale the investigation of bursting properties can be effectively tracked. The culture depicted is grown on glass and confined within a circular diameter. The culture contains about 3000 neurons.
b Bright-field image showing a detail of the culture and the distribution of neurons. The circle identifies a single neuron. **c** Corresponding fluorescence image during a spontaneous activity event. Bright spots are firing neurons. The resolution of the image is the same as the actual measurements. **d** Fluorescence signal from a 30 minutes recording of the spontaneous activity in the culture shown in **a**, averaged over the 500 brightest neurons. The top plot corresponds to measurements with both excitation and inhibition active (E+I network); the bottom one corresponds to excitation-only measurements (E network), with inhibitory synapses blocked with 40 μm Bicuculline. Fluorescence peaks identify network bursts. The symbols below each burst identify its initiation in a specific area of the culture.

oscillatory neurons which displays spike-driven adaptation, this result was proven on fully connected graph. The experiment conducted at Soriano Lab, UVB, Barcelona reveals that the waves nucleate at specific sites of the culture and propagate very rapidly (with an increasing speed during maturation), and then are followed by long periods of silence or very low activity. More importantly, these sites are specific (from culture to culture): there are preferred zones in the system where waves nucleate, and these are not correlated with any local properties of the network (firing rate, density, connectivity, clustering, ...)[5].
To solve the puzzling phenomenon of bursting is an essential step towards the realization of neuronal device and, more important, the understanding of neuronal encoding of information. In this perspective the use of activity simulators, e.g. NEST[6], assumes a fundamental role: it can help to test hypothesis and make predictions. Whether the dynamics of fully connected graph can be easily performed the reproduction of a realistic network has not be taken for granted, because the actual adjacency matrix of cultured network remains mostly unknown. In the next chapter we will see the important role NetGrowth can fulfill in this context.

2.4 Neural circuitry in Lab-On-A-Chip technologies

In this section I will report the work done by R. Renault during his PhD in the MSC Lab, focusing on the studies and experiments he accomplished towards the realization of an *in vitro* functional cell, also said a Lab-On-a-Chip.
R. has focused on shaping the networks at large scales, organizing the connectivity between groups of neurons instead of guiding single neurons. The organization of neurons on culture scale can be controlled by direct physical confinement; this is generally achieved using micro-fluidics chips. They consist in molded silicon polymer (polydimethylsiloxane or PDMS) covalently attached to glass coverslips through plasma- activated bonding. He proposed a new paradigm for neuronal devices: Primarly, microwells were used to compartmentalize cultures into neuronal populations, individually acting as super-neurons. These super-neurons are both robust and customizable as their properties emerge from the mean composition and density of the seeded cell suspension, which can be controlled accurately. They are also easy to read, as neuronal populations can display all-or-none bursts of activity which are easily detectable. Even if the amplitude of such bursts can vary and therefore carry information relevant to neuronal computation, they can also be seen as stereotypical events corresponding to super-neuron spikes; the investigation of temporal codes is therefore straightforward in such scheme. Secondly, the populations are connected through arrays of axon-selective micro-channels which control topographically the initial connectivity between them. Since the directed connections in the network are spatially separated, each can be reinforced individually without affecting the others.
This type of architecture can bypass the limitations of previous approaches where learning is hindered by highly reciprocal and overlapping connections in the culture. Additionally, network topologies developed in the field of ANNs can be easily

[5] Javier Orland website
[6] http://www.nest-simulator.org

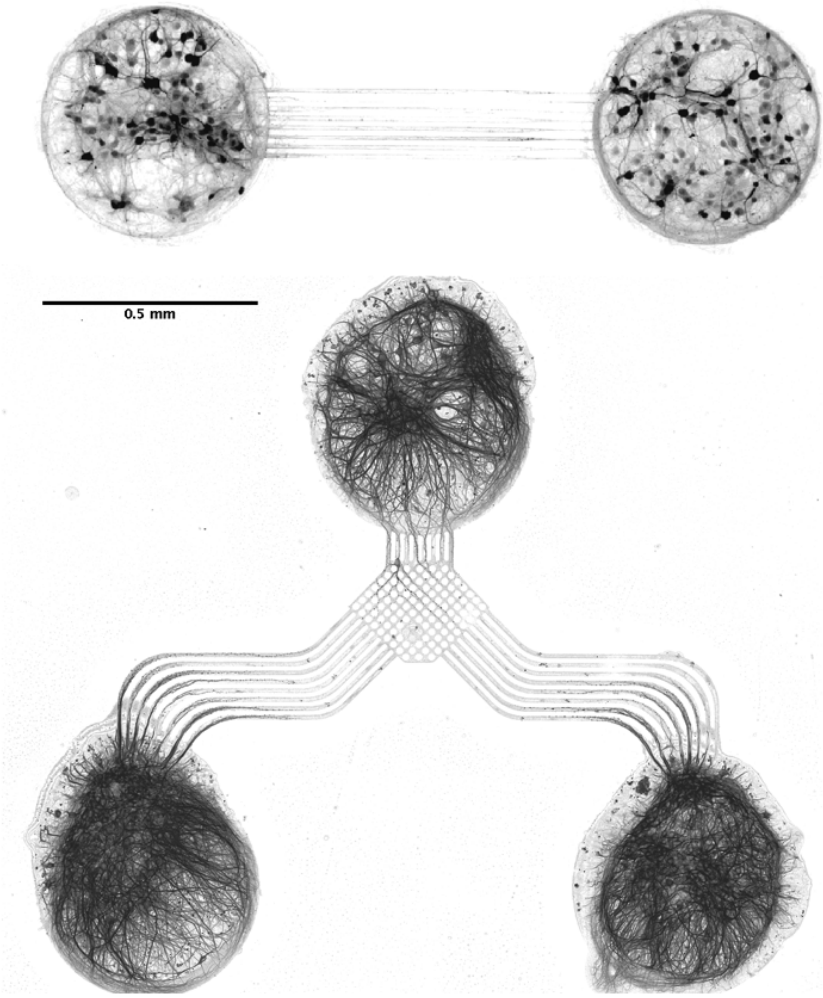


Figure 2.4. Minimal neuronal devices Neuronal devices helps investigating spatial and time coding of neuronal networks. The paradigma proposed in [Renault et al., 2016] uses large population as super neurons, whose activity pattern should be more reliable. Populations are connected with directional tunnels as those connecting the population in the pictures. These tunnels can be diode like or not, the axon is expected to walk up these structures looking for post-synaptic targets.
Top: Two populations of circa 700 fluorescent neurons are connected through micro-tunnels, the geometry of which can be controlled to yield unidirectional connectivity. Only a fraction of neurons are fluorescent here. **Bottom**: Three populations device (two inputs and one output) showing neurons stained for the cytoskeletal protein MAP2. Two and three populations devices constitute the building blocks of more elaborate devices.

implemented using this approach, see Fig. 2.4. The fact that the network topology inside each population is not constrained in such devices means that a lot of computational power is potentially lost. In the worst case scenario, and provided that populations can actually act as super-neurons, the number of useful computing units would be the number of populations, which is far less than the actual number of neurons. However, random sub-circuits inside each population could be exploited to multiplex different functions on each unit, as it is done on unconstrained cultures growing onto MEA. The computational capabilities of population units are therefore difficult to predict, but they would at least support the properties of the units inside a perceptron.
The learning protocols, Renaud carried through, failed for most of the structures where neurons were cultured, but the idea stays and always more groups and projects are studying the possibility of complex and confined structures in cultures. The costs of creating a functional cell, from the realization of the PDMS structure up to the learning protocol, unveil the remarkable necessity of creating a cheap test bench. The simulator is, in this case, a suitable instrument to explore the fog and test culture shapes and learning protocols. Furthermore in the context of axon guidance the availability of a modular tool, to implement and verify chemio-mechanical interactions model, is very handy as well. In the next section the simulation of culture is detailed, and some other cases which can benefit of NetGrowth simulator are indicated.

Chapter 3

In silico neuronal cultures, applications and limits

Whether the complexity of life beings, the multitudes of chemical reactions and micro-physics process go far beyond the computational possibilities of modern computers[1] the computers have revealed powerful tools at the service of neuroscientists [Amit, 1998]. Nowadays each research lab has an IT infrastructure, many have computing servers, and very often researchers are skilled in one or more programming languages. The possibilities opened by the spread of IT technologies are numerous:

- Non analytical theoretical models can be implemented and checked, e.g. simulate the propagation of an electrical signal on a complex dendritic tree.
- The research can be performed on synthetic data, avoiding the complex and expensive infrastructure needed to grow the neurons, nor the expertise to perform the experiments.
- The results become very reproducible and can be tested by the scientific community. Other researchers can recreate the experiment and adjoin new research pathways.

In this section we will zoom on the problem of reproducing neurons and neuronal cultures with computers. In the first section I will have a look to the perspectives and applications of neuronal simulations, stressing their relation with *in vitro* experiments. Then a brief review of existent simulators will be presented, focusing on those most related to the NetGrowth simulator.

3.1 Why simulate neuronal growth

There are many reasons why creating effective and realistic neuronal morphologies can be interesting; they come from both neurobiology and theoretical neuroscience. This double importance is the consequence of the many scales observed in neuroscience. Let's introduce these quests and show how the simulation of neuronal

[1] It actually depends by the scale of the process we want to reproduce, indeed the folding of proteins can be simulated, either, but once at time and for just few seconds

morphology paves the way to a deeper understanding of both neurobiological and cognitive processes.
Regarding the former, the growth of neurons, and the various morphologies they assume, is a puzzling question for biologists and biophysicists. The mechanisms that regulate the growth, both chemical and mechanical, are a subject of significant interest and experimentation in modern research, as I had the opportunity to understand during the Neuron Technology summer school. The study of the mechanical interactions between neurons and substrate opens the doors to relevant medical applications, e.g. in the field of rehabilitation. The resulting morphology sheds light on the chemical process going on below the cell surface and accounts for the role of specific neurons in the population: its ability to integrate signals and process information depends mainly from the dendritic tree. All this variable are kept into account in biophysical models as those presented in [Recho et al., 2016], or [Hely, 2001]. Neuronal growth simulators offer the opportunity to test these models and observe the consequences over the whole morphology, opening the door to effective comparison between theoretical model and experimental evidences, i.e. the morphology observed and catalogued in morphology databases as Neuromorpho[2].

When we are looking at the neuron from the top of cognitive basic functions, like spatial or working memory, the morphology appear much less important. Its relevance is covered by the multitude of neurons we are looking at, but it is the fundamental process that provide for the establishing of a neuronal network. Whenever the researchers are interested only to the adjacency matrix of the network, neglecting the synapses localization and the neurite extension, the simulation of realistic morphology is necessary. The function of the graph, and the macroscale properties of the network depend by the number of connections a neuron successfully establishes. A more concrete example is showed from three relevant publications on the shelf of culture activity studies: all the articles use a growth model to establish connections and generate the adjacency matrix of the culture. In case of [Orlandi et al., 2013b] the neuron morphology was almost absent and the connections between neurons were established with a very abstract method, this is detailed in Fig. 3.1, the generated network was used to simulate the culture and the Calcium Imaging measures compared with those *in silico*. The second example is more accurate: starting from NetMorph[3] simulator, the authors have studied the variation of synapses density, during the outgrowth of environment independent neurons. The effectiveness of this simulations cannot be proven directly, but the idea is more empirical: since it produces plausible independent morphologies the juxtaposition of many of them will produce something acceptable.
Eventually the project in [A Gritsun et al., 2012] starts from the NetMorph simulator to build a more consistent network: here the growth cones are driven by chemotactic gradients and the synapses are computed over density estimation of neurites' fields. Some details are discussed in Fig. 3.1.

In the next section some of the available and published simulators will be described, with a short description of pros and cons.

[2] http://neuromorpho.org/
[3] described in next section

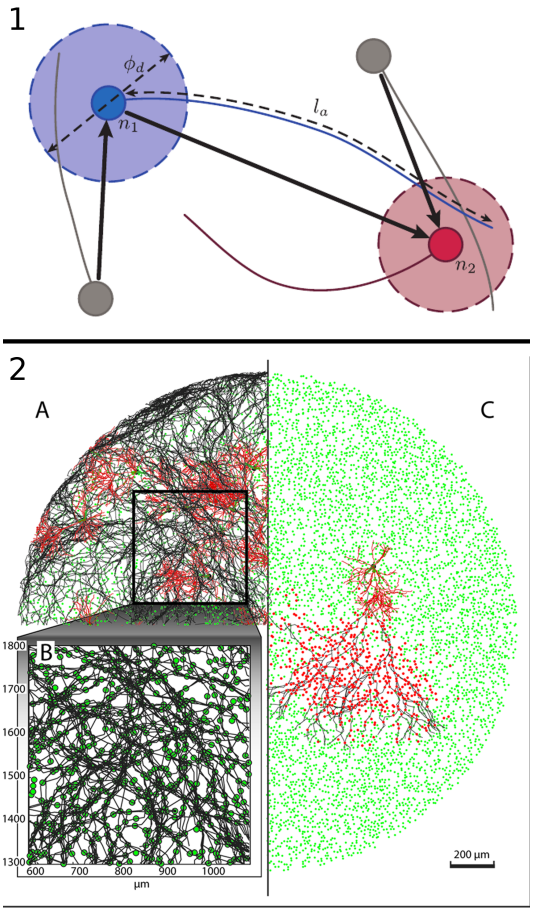


Figure 3.1. morphology like simulator for *in silico* network edification Creation of artificial networks of biological neurons *in silico* requires a heuristic method for synapses creation. In picture **1** Soriano, Orlandi & co.[Orlandi et al., 2013b] creates a neural connection whenever the extruding axon l_a intersects the neurite circle, ϕ_d is the neurite density. This approach is simplistic but very fast and allows for creation of huge networks ($>$ 5000.000 units).
Picture **2** present the network generated in [A Gritsun et al., 2012]. It is NetMorph based and implements the chemotactic based drift. The network is very huge in this case and the neurites' morphology is plausible, unfortunately the code was not available on the web and generally it presents the flaws of NetMorph simulator, it is very hard to hack.

3.2 State of the art in neuronal growth simulation

The history of neuronal growth simulator has evolved in the past twenty years, first examples were related to the publication itself and the implementation was very basic, usually the model implemented was detailed in some aspects and neglect some others. After more sophisticated model were proposed the simulator started to grow in complexity and took into account the branching functions, the environment and so on. A complete review of the active simulator could appear as a boring enumeration, I have preferred to select the relevant aspects of each one and stress the new possibility the simulator was offering.

One of the first simulator I found in the bibliography is the one devised by Segev and Ben Jacob in [Segev et al., 2017]. They provide a model for the **self-wiring of neuronal networks**, based on chemotaxis. Their model contains migrating growth cones without detailed neuronal morphology. The simulation is separated in sequential steps:

1. Each unit *cell* releases a chemo-repulsive agent and sends neurites one at a time.

2. All newly formed walkers *growth cones* are initially sensitive to the chemo-repulsive agent.

3. After reaching a specific length, the walker switches from sensitivity for the chemo-repulsive agent into sensitivity for the chemo-attractive agent. The walker also emits an amount of triggering agent.

4. The specific length of the neurite is determined by the dynamics of the walker's internal energy which is controlled by the soma.

5. The triggering agent diffuses in the media. When the triggering agent concentration exceeds a certain threshold, cells in the neighborhood respond by emitting the chemo-attractive agent.

6. Subsequently the walkers move to the attractive response. This scheme is very simple but effective; the interplay between chemoattraction and chemorepulsion may be keystone in the understanding of network formation. The investigation proposed in the article are indeed very close to the ones of NetGrowth project, but twenty years of technological improvements make the simulator obsolete and was impossible to rescue the code. Furthermore the morphology was absolutely absent in this simulator.

A completely different approach is pursued from Luczak in [Luczak, 2006] : The main objective of this work is to illustrate that the creation of complex reproducible dendritic trees does not require precise guidance or an intrinsic plan of the neuron geometry. Rather that external factors can account for the spatial embedding of the major types of dendrites observed in the cortex. In this model the number of terminal branches, the mean and maximum branch orders, and the fractal dimension and other parameters of dendrite geometry are all controlled by a few basic environmental factors. The most important factor in determining the shape of generated neurons is the space available for growth. The neurite develops by a **Diffusion-Limited Aggregate model**, described in Fig. 3.2 plus some ad hoc constraints like spatial limitation, pruning and competition. Whether this model reproduces strikingly plausible dendritic tree the axon is never mentioned and we believe the same model will not be sufficient to reproduce the axon outgrowing observed in vitro experiments.

This model is however very relevant to stress the importance of spatial confinement and spatial interactions among branches of the same neurites.

Just a few years after Randal Koene and colaborators coded **NetMorph** [Koene et al., 2009], which is, up to now, the state of the art simulator. It is based on fully generative models, developed by the group of Van Oyen and Van Pelt in Amsterdam. NetMorph cannot deal with spatial confinements and produces only 3D morphologies, the 2D simulator exists but does not work very well. The merit of NetGrowth was to produce neuronal morphologies based only on internal processes, which are usually derived by a mathematical modelling of experimental observation. This peculiarity is also a weakness, indeed the importance of environment interaction was steadily demonstrated with lab's evidences. This time the code was available but was impossible to fit it to our scopes, the code was intrinsically not modular and we have not find any effective solution to upgrade it to our solutions. Furthermore it was completely coded in C++, with a not welcoming interface. NetMorph will be an important touchstone for our project and we hope to be able to reproduce its results and even go beyond. Eventually some years ago Torben Nielsen and collaborators proposed a new framework [Torben-Nielsen and De Schutter, 2014], while the simulator was never finished, the idea of using a full parallelized paradigm for neuronal growth stays. Previously presented simulators were able to produce just a few of neurons with reasonable resources, instead with **NeuroMac** the growth of an entire culture was possible on personal computers(easymotion-prefix). A sketch of NeuroMac an NetMorph is offered in Fig. 3.2.
To conclude this paragraph let's stress some overall properties of the presented simulations:

- All these simulators are generative these means that produce the morphologies from scratch, without any statistical mix of observes-reconstructed morphologies
- None of the simulator proposes to study the interaction between neuron and mechanical cues, indeed the Mechanobiology is a very recent research path. It is linked with the technological improvements of last years and is a relatively unknown field.
- Only NetMorph is coded with a user interface in mind, others are mostly linked to a specific research project. Conversely it is coded with a rigid infrastructure which does not allow for an upgrade towards growth-activity simulation.

The overall flaws of each simulator is the impossibility to reproduce the exact dynamics of the living matter. A simplification of the process is necessary to enlarge the scale of the simulation, this problem is an open issue of computational approaches and the possibility of a deeply multi-scales simulation is just a perspective for the current state of Bioinformatics. We hope the IT structure of NetGrowth, which is thoroughly modular, will strike a blow in this direction.

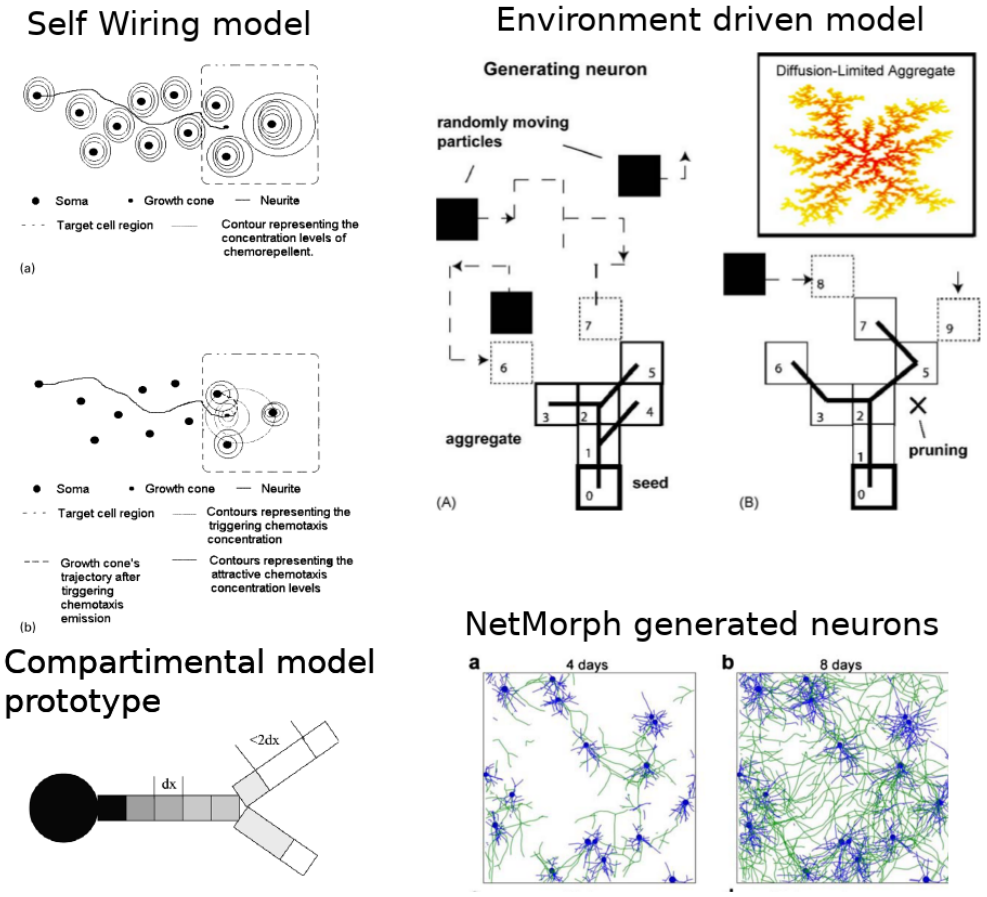


Figure 3.2. Simulator of neuronal growth and network's wiring Simulate the neuronal growth offers many possibilities to study both the growth process and simulate activity on large scales. The quest for the graph structure of neuronal networks is part of the general investigation on information processing in the nervous tissue. Simulators focus on different aspect of growth process, intrinsic forces or external stimuli are reproduced to compare the difference between model and experimental observations. The four simulations reported in the picture describe four different framework of synthetic morphologies. The self-wiring models focuses on creating the relevant connections among neurons, neglecting the morphology of neurons. Inversely the **Environment driven model** edifices the dendrites following the particles diffused in cellular matrix and studies the originated trees. NetMorph is the most similar project to NetGrowth and implements branching and navigation models to build independent neurons.
The compartmental model is a method to integrate the diffusion of a chemo-electrical signal throughout the neurite, it is a conventional method to integrate differential equation; the number of equations is set by the number of compartments, each compartments act as unit point and solve a discrete differential equation, the rate of out-coming and incoming material depends on adjacent segments. This method is not implemented in NetGrowth becouse the arising system is too much expensive from computational point of view.

3.3 NEST: the activity simulator

To conclude this brief review on state of the art I have to cite the most widely used simulator for *in silico* studies of network dynamics. NEST is a simulator for spiking neural network models that focuses on the dynamics, size and structure of neural systems rather than on the exact morphology of individual neurons. The development of NEST is coordinated by the NEST Initiative.
NEST is ideal for networks of spiking neurons of any size, it can help to test models of information processing and network activity dynamics, e.g. laminar cortical networks or balanced random networks. One of the research edge investigation using NEST is in the field of learning and plasticity of networks.
NEST is thoroughly parallel and Tanguy Fardet has devoloped a functional module, NNGT to deals with graph structure and other topological information. This tool is perfect to bridge our research project with NEST and use it to develop a framework of growth-activity simulation.

3.4 NetGrowth, a brand new neuronal simulator

In this section I will present the NetGrowth simulator, I will describe the collaborations from which it is promoted and point out the new elements it introduces in the panorama of neuronal growth simulators, an example is presented in Fig. 3.3
NetGrowth is a parallel simulator of cultured (2D) neurons, it simulates the growth process, from the first phase of neurite differentiation up to mature phase, a period of about three weeks. The neurons can interact with the environment through their growth cones, the neurite extruding tips. The user can set models and parameters of the simulated cells; the parameters can be changed during the growth process itself, and, these features permit to reproduce empirical observation that are not comprised in any model. All the models are correlated with biological evidences and biophysical models. The environment interaction objects to reproduce the *fasciculation* between neurites and the formation of synapses, based on the dendritic spines evidence. The informatics infrastructure of NetGrowth reproduces closely the one from NEST, the aim of the project is, indeed, to complement the NEST simulator, towards an integrated simulation of activity and growth.

The NetGrowth project is the result of the collaboration among the Neurophysics group at MSC Lab, the group of Catherine Villard at IPGG, Paris, the group of J.Soriano at UVB, Barcelona and E.Moses, Weizmann Institute, Rehovot. The research groups are involved in different stages of neuronal cultures studies, from the axon guidance up to the implementation of learning protocols for the neurons' population. I had the chance to discuss about NetGrowth project with the group of Villard and Soriano. They represent two of the future, possible, users of the simulator: the former is looking for *in silico* studies to compare the confinement strategies for neurons. The latter needs realistic network topology and morphology to base the NEST simulations. I had the opportunity to visit Catherine Lab and I plan to visit Soriano Lab in May, and, for the occasion, help them to realize the simulation of a new culture PDMS structure. The interaction with these groups is a

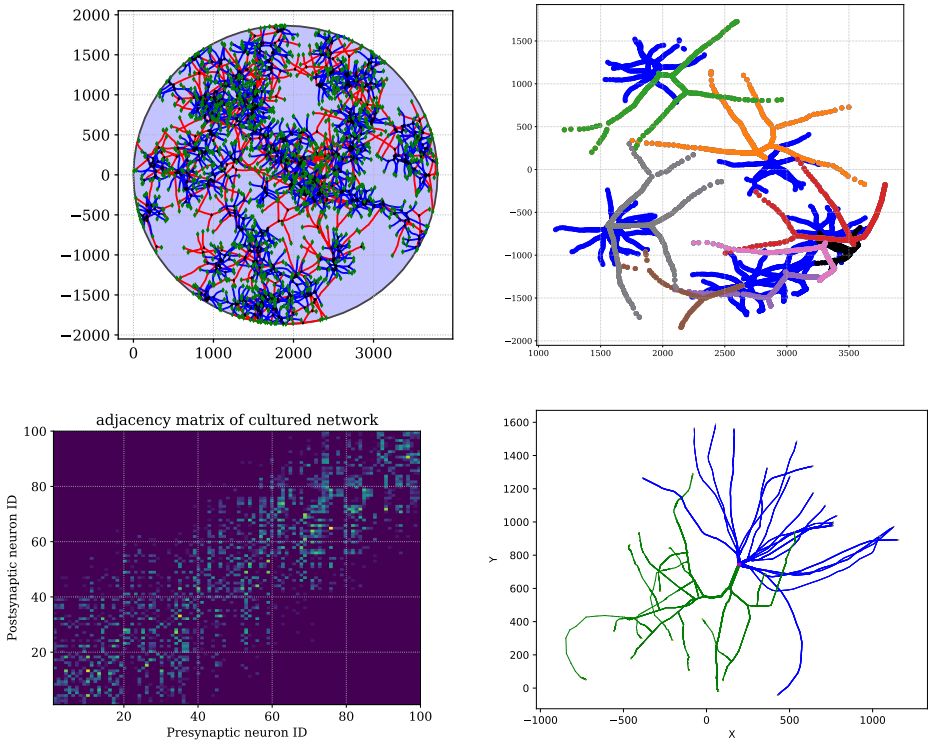


Figure 3.3. A sample culture generated with NetGrowth NetGrowth simulator can simulate biologically plausible neurons in confinement. The example presents 100 neurons generated in a Petri dish-like culture **Top-left**. Neurons interact with the environment and create complex shapes, merging the forces imposed by the mechanical traction with the internal attitude to grow out from the soma **Top-right**. It is possible to evaluate whether two neurons intersect or not and hence create the adjacency matrix for the simulated network **Bottom-left**. When the neurons are not constrained they assume the habitual morphology, e.g. the neuron in **Bottom-right** presents a Chandelier like morphology, generated with models discussed in the manuscript.

splendid example of coordinated research: different experiences and competences are joined together at distinct level of the investigation chain, in order to obtain more reproducible and consistent results. In this frame we fulfil the role of theoretical physics and bio informatics, translating experimental evidences into feasible models, that can be used for larger scale investigation. On the other hand the group of C. Villard, in the person of Ian Audric, has furnished relevant and irreplaceable data whom the simulator is based on.

With the NetGrowth simulator, and the NNGT module developed by Tanguy Fardet, the scientific community of neuroscience is enriched of a sophisticated and performant tool. Starting from NetMorph, which is the state of the art in 2D and 3D growth simulation, our project adds consistent improvement. From the scientific point of view the insertion of mechanical interactions rules, the possibility of confinement and, in soon, of fasciculation and chemiotaxic guidance is a significant step forward. This improvement responds to the technologically advancement in the realization of micro-fluidic chambers. On the hand of technical features it presents a change of paradigm: NetGrowth has a parallel infrastructure that responds adequately to IT resources and trends of today and tomorrow. Since T.Fardet, who I collaborated with in the developing of the project, has a role in the Nest developers and users community, the similarity to the *de facto* simulator of neuronal activity are numerous. We hope to integrate the projects in the next future, opening the door to very new perspectives: an efficient simulator of activity and growth, where the electrical stimulation influence directly the outgrowing properties and vice versa. Eventually the simulator is realized in favour of a broader community of users in respect to its predecessors: while the efficiency of the compilation is granted from the C++ language, with which the models where implemented. The simplicity of use is offered by the user interface written in Python, the most used language in research labs.

Part II

Neurites outgrowth in spatially constrained environments

This part of the thesis will deal with the NetGrowth simulator. In the first chapter the biological process of neuronal growth will be described. Then we will recall the biophysical models in scientific literature and eventually discuss the models implemented in NetGrowth.

Chapter 4

Introduction to biophysics of neuronal outgrowth

Contents

In this chapter the preparatory work done for the NetGrowth simulator is presented. The chapter is divided in two sections. In the first we focus on the growth cone behaviour, looking at the interactions with the environment, the steering models, and the elongation rates. The second section gives a hint of the complex world of neurite's branching.
But first, a short description of neuronal growth and its several stages is offered before entering in the details of growth cone biology.

Neuronal growth stages Let's briefly describe the stages of neuronal outgrowth to familiarize ourselves with this very complex biological process.
During culture lifetime, neurons will extend their neurite and create synapses with other neurons. This process takes about 21 days. At this stage of the project we are considering the neuron fixed on the substrate by the use of some adhesive proteins.

Neuronal growth occurs in several stages:
first, a *neurite initiation* process occurs in which hand-like extensions of the cell membrane – the lamellipodia – develop into short neurites with a rich actin frontier. This process is enhanced by the actin-waves.
After a period of outgrowth and retraction of these newborn neurites, one of them develops the most and elongates rapidly over 2 days while inhibiting the outgrowth

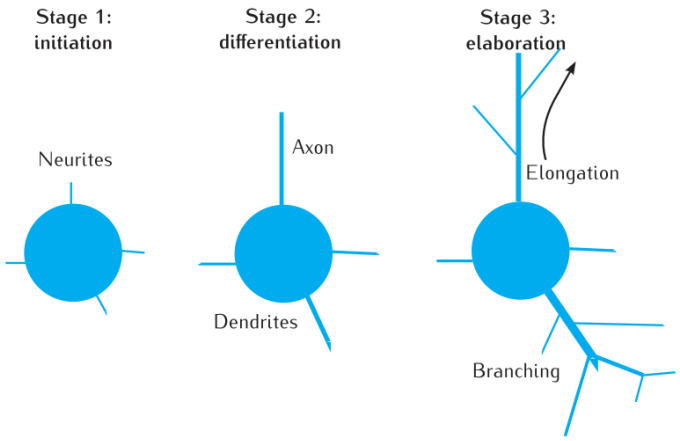


Figure 4.1. Stages of neuronal arborization The growth process is composed of three main stages of neurite development: (1) initiation, (2) differentiation into an axon and dendrites, (3) arborization, including elongation and branching. The whole process lasts 3 weeks. Afterwards, the growth slows down and a reshaping phase initiates during which most of the dendrite is pruned. The topographical plasticity slowly disappears in favour of synaptic plasticity.

of the remaining neurites. This predominant neurite becomes the axon and assumes distinct biological characteristics: the neurites now have *differentiated* themselves into axon and dendrites.
The dendritic arbor comes out after a long period of **elongation and branching** of each neurite. At first the arbor grows rapidly with respect to the total neuritic length, and the structure changes repeatedly. Eventually, the elongation rate slows down and the global shape of the arbor becomes stable.
The presence of obstacles, mechanical cues, or chemical ones drastically changes the shape and the timing of the arborization. The Growth Cones (GC) either show a repulsive or attractive behaviour with respect to different external stimuli. Such properties can be used to drive and control neuronal network formation.

4.1 Growth Cone: searching for biological targets

The Growth Cone is one of the most amazing actors in the outgrowth process of neurons. It is the tip of the neurite. It senses the environment and assembles tubulin in micro-tubules in order to elongate towards its biological target. A simple sketch is offered in Figure 4.2.
To describe all aspects of Growth Cone elongation and retraction would require a long discussion. Moreover, no description would be completely exhaustive, since there are many diverging opinions. The interested reader will find a detailed review in [Franze and Guck, 2010] and in [Lowery and Vactor, 2009].

Here, we will give an overview of the principal forces acting on the stage. The goal is to develop a simple model for the Netgrowth simulator. Following the review in [Suter and Miller, 2011], we're going to present the recent studies on the elongation

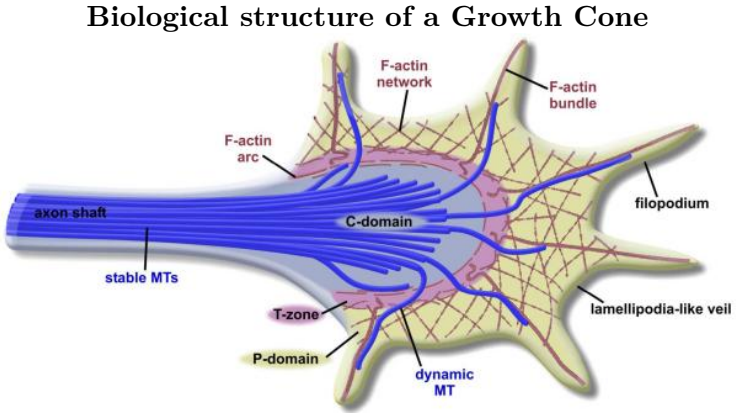


Figure 4.2. The structure of the growth cone is fundamental to its function. The leading edge consists of dynamic, finger-like filopodia that explore the road ahead, separated by Lamellipodia-like veils – sheets of membrane between the filopodia (see the figure). The cytoskeletal elements within the growth cone underlie its shape. The growth cone can be separated into three domains based on cytoskeletal distribution14. The peripheral (P)-domain contains long, bundled actin filaments (F-actin bundles), which form the filopodia. It also contains mesh-like branched F-actin networks, which give structure to lamellipodia-like veils. Additionally, individual, dynamic, and 'pioneering' microtubules (MTs) explore this region, usually along F-actin bundles. The central (C)-domain encloses stable, bundled MTs that enter the growth cone from the axon shaft, in addition to numerous organelles, vesicles and central actin bundles. Finally, the transition (T)-zone (also called T-domain) sits at the interface between the P- and C-domains, where actomyosin contractile structures – called actin arcs – lie perpendicular to F-actin bundles, forming a hemicircumferential ring within the T-zone33. The dynamics of these cytoskeletal players determine the growth cone shape and movement during its journey. Images from [Lowery and Vactor, 2009]

.

of axons, their response to traction, and the influence of the substrate on rate and direction of the elongation. The approach will be bottom-up, starting from the vary basic components of the growth cones up to the physical and computational models hopefully reproducing the elongation and the directionality of the growth cone.
For the following sections I thank Ian Audric, who has written a precise review on growth cone internal biology, for his work at IPGG in the group of Catherine Villard [1].

4.1.1 A pointed Little Head: biology of a growth cone

The growth cones are composed mainly by 3 regions, the *Central*, *Transition* and *Peripheral* domains. The first domain is dominated by the presence of microtubules while the rest by *lamellipodia* and *filopodia*,two actin filament structures. The macroscopic region containing the P ans T domains is called *kinetoplasm*, or Growth Cone, while the *axoplasm* connects the GC to the soma.
The equilibrium between these regions, the propensity to branch, the microtubule assembling rate and the direction are regulated throw hundreds of intracellular protein and extracellular signals, sensed and mediated throw the filopodia, the most important process in this respect is the turnover, that is the process in which the protein are polymerized into microtubules or actin filaments or depolymerized into molecular tubulin or G-Actin. Let's get into details describing the main actors in the outgrowth process.

The cytoskeleton is a very dynamic network of protein filament. It is responsible for sustaining the structure of the cell. The growth cone is a particular arrangement of cytoskeletal filament, protruding from the main cellular body.
The neurite presents three main polymers: the F-actin, the micro-tubules, and the neurofilaments [Coles and Bradke, 2015]. The first two dominate the short time scale which we are interested in when looking at the dynamics of the growth cone.

The actin filament (F-actin) is a semi flexible polymer organized in a double helix structure. As explained in Figure 4.3, the balance of polymerization and depolymerization is oriented. The actin is widely present in many cellular processes and it has the ability of building chains of forces. It is fundamental in processes such as cellular motility or mitosis. It can be organized in a dense network or in (anti) parallel bundles. See Fig 4.2.
In the axoplasm F-actin is mostly organised into a cortical mesh around the microtubule's inner core. The presence of myosin II motors in this cortex leads to the presence of contractile stresses [Jülicher et al., 2007]. In the GC, F-actin is organised in a lamellipodium similar to the ones found in cells specialised in crawling.

The microtubules (MTs) are rigid polymers of a diameter of around 25 nm composed of tubulin,ì – a proteic dimer. The tubulin is arranged laterally to create cylindric and hollow MTs. The MT has a persistence length of the order of millimetre, as defined and measured in [Martin, 2013]. Its stiffness gives a robust skeleton to the whole cell. It oscillates continuously between a phase of elongation and retraction, also known as catastrophe. The active equilibrium is finely regulated by a protein called MAP (Microtubule Associated Protein).

[1] http://neurofluidics.org/program/

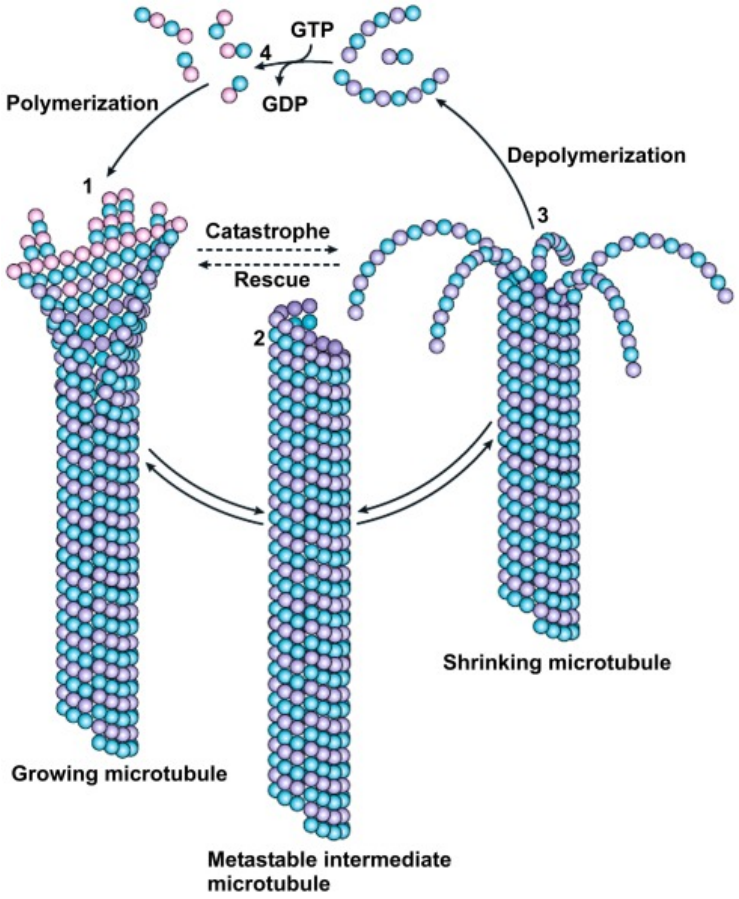


Figure 4.3. Microtubule polymerisation dynamics Cartoon representation of microtubule growth and catastrophe. For a growing microtubule (left), polymerisation occurs most rapidly at the plus (+) end through the addition and removal of$\alpha - \beta$-tubulin heterodimers. After incorporation into the microtubule filament, hydrolysis of the β-tubulin-bound GTP takes place. End-binding (EB) proteins can transiently bind to the plus end, influencing polymerisation dynamics both directly and indirectly by recruiting other important regulatory +TIPs. For uncapped filaments, slow polymerisation and depolymerisation can also take place at the microtubule ebottomnd, most likely through similar mechanisms as those occurring at the upon end. A reduction in the concentration of free $\alpha - \beta$-tubulin heterodimers, for example due to stathmin-mediated sequestration, is one way in which microtubule catastrophe may be induced (right), leading to filament shrinkage.

They can be cross-linked by molecular motors (e.g. myosin II for F-actin, Dynein or Kinesin for microtubules) able to exert active stresses inside the meshwork. Most of microtubules lie in the axoplasm and are connected by passive cross-linkers (MAP) that generate a network with a quasi-lattice structure. These microtubules are highly stable with turnover duration of hours, possibly thanks to the presence of MAPs.

The research is active on this topic and technological developments allow biologists to go further in the understanding of the complex behaviour of cytoskeleton polymerization and the roles of the many proteins active in the neurite filaments [Von Der Ecken et al., 2015].

Filopodia and Lamellipodia The growth cone contains an actin cytoskeleton which adds mechanical strength to the cone keeping its shape and as well helps driving and guiding the cone's movement. Filopodia are small actin filament bundles extruding from the tip of the growth cone. They grow and retract at a high rate constantly probing the environment and picking up guidance cues[2]. Filopodia can adhere to the substrate, producing tension within the growth cone [Mogilner and Rubinstein, 2005, Betz et al., 2011, Craig et al., 2012].

In [Athamneh et al., 2015b] the author offers a review over the many roles forces play during neurite outgrowth. How filopodia sense the environment is a relevant open question in this research field. Despite the lack of understanding of the mechanisms underlying mechanotransduction, experimentalists are using recent techniques – such as optical tweezers – to characterize the response of filopodia to external mechanical stimuli. Fig. 4.4 shows an experiment with a coated microneedle, in which experimenters measure the force exerted by the Growth Cone on a microneedle.

4.2 Elongation, steering and external stimuli

Elongation is a key feature for neurites' outgrowth. The Growth Cone senses the environment during this process. Its ability to act depending on the local properties and the external stimuli, is both the most relevant ability of this biological micrometric walker and the main limit for the neuron's growth simulators.
The direction of elongation and its intensity allow or not the GC reaches the target and fulfil its biological aims. This ability of navigating onto the substrates and reach its target is widely studied in biology labs [3].

The desire of controlling elongation and enhancing the elongation process *in vivo* is very ambitious and offers a wide spectrum of applications, mostly in medicine and rehabilitation. Still, the mechanism of finding the axon path is really complex, with hundreds of chemical [Wen and Zheng, 2006] and mechanical [Chua et al., 2014] cues interacting with a biological and active walker! Fortunately, the development of bio-mechanical technologies, as the AFM (atomic force microscope) or the optical tweezers, are offering a wider spectrum of measures. They are crucial to understanding this process and investigate the mechanism underlying the complex behaviour of

[2] Many videos are available on `youtube.com`.

[3] `http://neurofluidics.org/program/http://phdneurobiology.sissa.it/eng/faculty/laura-ballerini.aspx`
`http://phdneurobiology.sissa.it/eng/faculty/vincent-torre.aspx`

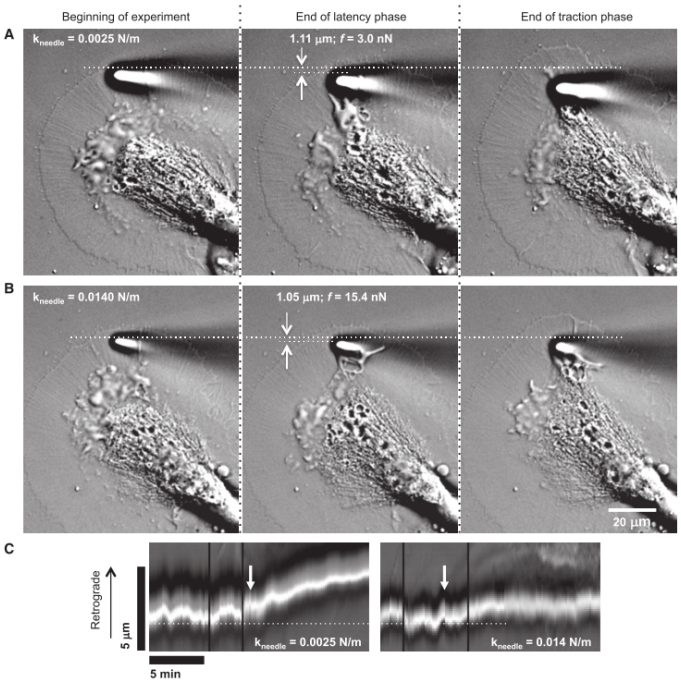


Figure 4.4. A biomechanic experiments to measure traction forces The growth cone produces different amounts of traction force during adhesion-mediated advance. (A) DIC images of a growth cone interacting with a 0.0025 N/m Con A-coated microneedle at the beginning of the experiment, at the end of latency and traction phases. (B) The same growth cone shown in (A) interacting with 0.0140 N/m Con A-coated microneedle. Needle stiffness (k), needle deflection, and traction force are indicated. (C)Kymographs showing the position of the microneedle throughout the time course of the experiments shown in (A) and (B). Arrows indicate the end of latency phase. Images from [Athamneh et al., 2015a]

axon and dendrites elongation.
We will show how the forces – intended as mechanical pressures and pulling – are getting a relevant place in biology and neuroscience and how the possibility of quantify them is changing models and approaches [Athamneh et al., 2015b]. Oversimplifying the process, we can account for the elongation with the synthesis and fixation of tubulin in the micro-tubule formation. The steering can be assumed as the direction where such microtubules grow longer, as in Figure 4.5. Most diagrams of axonal elongation are, in essence, models of microtubule and actin dynamics occurring in the growth cone as it grows in response to extracellular cues. The debate is still open on which between the pulling of actin cytoskeleton and the pushing from the back plays the most significant role [Rauch et al., 2013]. However, this simple scheme hides most of the open questions in modern research on growth cone. In a recent article [Kahn and Baas, 2016] the author wondered whether the microtubules play the main role in the invasion of the peripheral domain and in the consequent turning, or if the actin cytoskeleton is alone responsible of Growth Cone steering. This leaves us with open questions as:

"What are the means by which each relevant molecular motor protein is activated or deactivated in a manner that contributes to the turning of the growth cone in response to environmental cues?"

The leading interpretation describes the process in the following manner. Polymerization of actin filaments at the periphery of the growth cone and central depolymerization effectively result in actin bundles moving backwards through the growth cone, giving the growth cone a caterpillar track style movement. The space that is created behind the growth cone by this pulling action is filled by microtubules and elongation occurs [Kiddie et al., 2004].

The complex mechanism by which the flux of actin determines the synthesis of microtubule is too detailed for the scope of this work, the interested reader can refer to [Bornschlögl et al., 2013] which offers a purely mechanic model, synthesized in Fig. 4.6. Following the excellent work of P. Recho in [Recho et al., 2016], it is possible to distinguish whether they imply that the GC pulls the trailing axoplasm thanks to F-actin polymerisation pushing the membrane in the P-domain and myosin II contractility pulling from the T- domain or whether it is rather microtubules which, from the axoplasm, polymerise against the T-domain and propel the GC. Even whit such basic knowledge on Growth Cone dynamics the relevance of interaction with environment come out. To study the reaction to mechanic or chemical stimuli leads to different research projects and questions, and the literature is very broad on the topic.

Let's now focus on understanding how the mechanical stimuli offered from a wall or from a generic stiff object will influence the behaviour of the growth cone. This subject is commonly designated as mechanical signalling. Even in this case the research is far from being concluded; the details how the different events are orchestrated to gradually build up traction force, up to hundreds of nanoNewton, and guide axonal growth in the direction of the adhesion site. In [Athamneh et al., 2015a] the authors show that traction force increases gradually over time as the growth cone encounters a new adhesion substrate. The maximum level of the generated force depends on the stiffness of the new substrate, implying continuous strengthening of

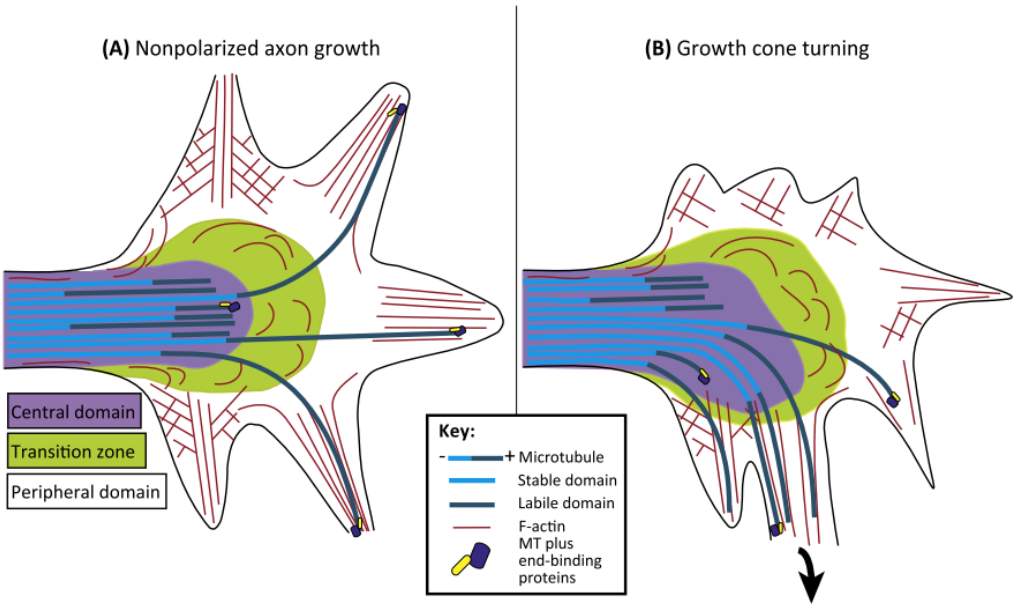


Figure 4.5. Scheme of Growth Cone turning The central domain of the growth cone is the microtubule (MT)-rich region contiguous with the axon shaft. The peripheral domain is the outer-most part of the growth cone. The peripheral domain comprises a broad flat lamellar region in which actin filaments are arranged as a meshwork, as well as elongated thin filopodia in which actin filaments are arranged as aligned bundles. The transition zone is the region between these two domains. Retrograde flow of the actin cytoskeleton in the peripheral domain pushes back most microtubules, compacting them in the central domain. Individual microtubules from the central domain are able to penetrate the transition zone to enter the peripheral domain during growth cone advance. (B) During growth cone turning, microtubules extend from the central domain through the transition zone preferentially in one side of the peripheral domain. Image from [Kahn and Baas, 2016].

Mechanical model for Growth Cone elongation

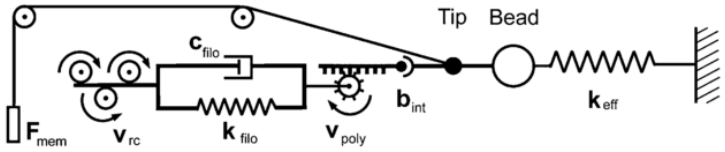


Figure 4.6. Mechanical model for filopodia. Force production via the tip arises from the parallel action of membrane forces (F_{mem}) and from actin dynamics. Cortical retrograde flow (v_{rc}) couples with high friction to the filopodial actin shaft, modeled as a Kevin–Voigt body with an elastic modulus k_{filo} and a viscosity c_{filo}. Polymerization at the tip (v_{poly}) slower than the cortical retrograde flow leads to retraction. Cytoskeletal force transduction can fail due to weak actin-membrane linkage at the tip (b_{int}). An elastic force with stiffness k_{eff} controlled by the optical trap is exerted on the bead. Image from [Bornschlögl et al., 2013].

the clutch and/or active recruitment of molecular motors. From previous studies it is also known that the growth cone response to a physically restrained adhesion substrate includes adhesion formation, SRC-tyrosine phosphorylation, slowing of retrograde F-actin flow, increased actin assembly, advancing of microtubules, and leading edge advance.

4.3 Growth cones in constrained environments

Here we discuss some recent experiments performed to study the interactions between the active growth cone and the physical environment. We have seen that the variety of behaviours and stimuli is vast; therefore we will focus on the experiences closer to our laboratory and to the aims of NetGrowth.

Experiment on environment interaction

Let's now discuss some recent experiments performed to study the interactions between the active growth cone and the physical environment, we have seen the variety of behaviors and stimuli is spread so the attention will be focused on the experiences closer to our laboratory and to the aims of `NetGrowth`.
When talking of culture over substrates, and the consequence of, a net example of mechanically enhanced system is offered by the work of Ballerini's group at Sissa. They study cultures growing on a substrate of carbon nanotubes (CNT) [Gangaraju and Lin, 2009]. Here, the presence of CNT enhances the growth of neuronal tissue and accelerate the stabilization of synaptic cross-links up to the standard connectivity reached in control cultures. In this case the signaling is proved to be not only mechanic but electric too. Some applications of this experimental setup were in a rehabilitation frame, helping the reconnection of bone ganglia.
The same group worked on 3D cultures built with transparent PDMS and studied the change the third dimension carries on topology and geometry of the network [Bosi et al., 2015], an example of 2D and 3D cultures in Fig. 4.7.

Another project we have collaborated with for the NetGrowth initiative is held by C. Villard and collaborators.
The group, located in the microfluidic lab of IPGG in Paris, is interested in measuring the modifications the mechanical environment produces on Growth cone navigation. A first and seminal experiment was carried by R. Renaud, whom this thesis dues very much. He has measured the affinity of growth cones with PDMS structure, revealing the formers' attitude to follow the structure edges. Such affinity has been evaluated numerically as explained in Fig. 4.8 The group is also working on the diode project discussed in Fig. 4.11, and in general on the modifications that a narrow environment acts on the growth cone behavior. They have implemented a selective "return to sender" strategy by exploiting this phenomenon, redirecting axons growing in the unwanted "reverse" direction to their original compartment via U-shaped lateral connections; and the efficiencies of different variations of this basic concept regarding their direction selectivity, and demonstrated that they constitute a major improvement over the tapered channels already used by the brain-on-a-chip community.

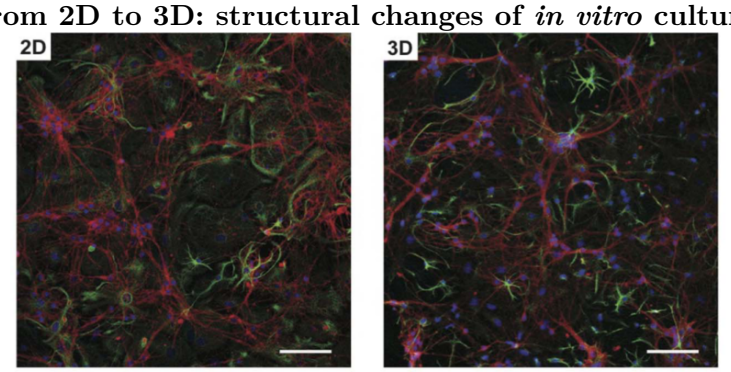


Figure 4.7. In (a) (top row) confocal micrographs show hippocampal cultures grown on 2D-PDMS (left) and 3D-PDMS (right) immune-stained for b-tubulin III (in red), GFAP (green) and DAPI (blue), scale bar is 100 mm. 3D platforms exhibits properties and behaviours that differ from their equivalent 2D configurations, neverthless neurons that grow on supports, such as multi-layered artificial substrates, are only in part exposed to the 3D environment, due to cell adhesion to the platform(s). It's stressed that 3D networks, besides displaying a similar degree of active cells when compared to 2D ones, display a higher rate of bursting, under both spontaneous and disinhibited conditions, due to an improved efficiency of neurons and neuronal connections in synchronizing their activity.[Bosi et al., 2015]

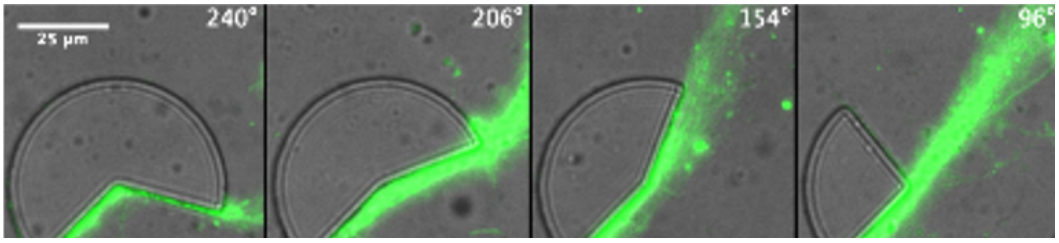


Figure 4.8. Growth Cone affinity to PDMS structures In this simple set of experiments R. Renaud studied the tendency of a Growth Cone to follow the edge of the PDMS structure when it suddenly turns of a certain angle. The growth cones are highlighted with a tubulin bonded green dye. To measure adequately the affinity and the percentage of adhesion is very hard with this setup, nevertheless it is possible to evidence a qualitative displacement from the structure when the angle is greater than π

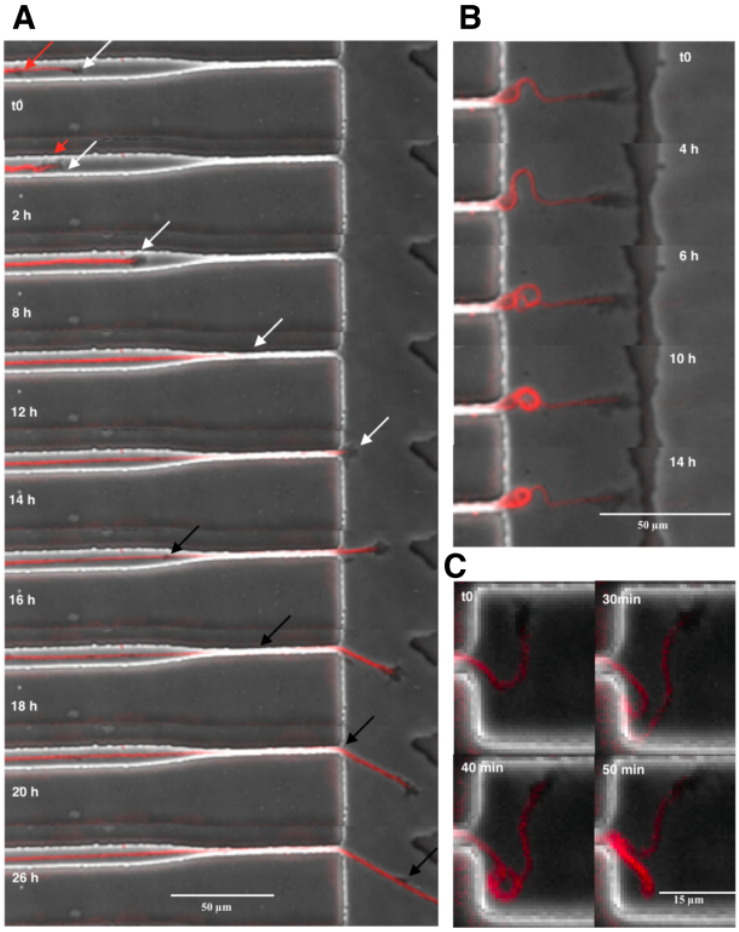


Figure 4.9. Axons grow in tiny PDMS microchannels, from I. Audric experiment at IPGG. (A) A growth cone advancing into narrow micro-channel. Between time zero and 2h the GC (white arrow) retract itself and pause until a following GC joins it (red arrow). They keep on going forward without further pauses. Another cone is visible at time 16h (black arrow). (B ; C) Self-rolling phenomena in GC encountering an obstacle.

In a recent work, Ian Audric, exposes the GC to a set of different shapes looking for common behaviors and aiming to a comprehensive description of constrained growth. Unfortunately biology is hard to forecast and the experiment ends up with a too poor evidence to asses any result. The author is working the project again and we hope the NetGrowth simulator will benefice by his work in next months. A picture of the experimental set up is offered in Fig. 4.9. [p]

As I have remarked until now, the bibliography is really huge and I have done a big effort to summarize the relevant aspects, both in this relation than in the research work itself.

I hope the reader have a reduced but clear understanding of the neurite outgrowth and growth cone navigation to comprehend the next sections, where biology effectiveness is put aside for more affordable and computable models of neuronal outgrowth.

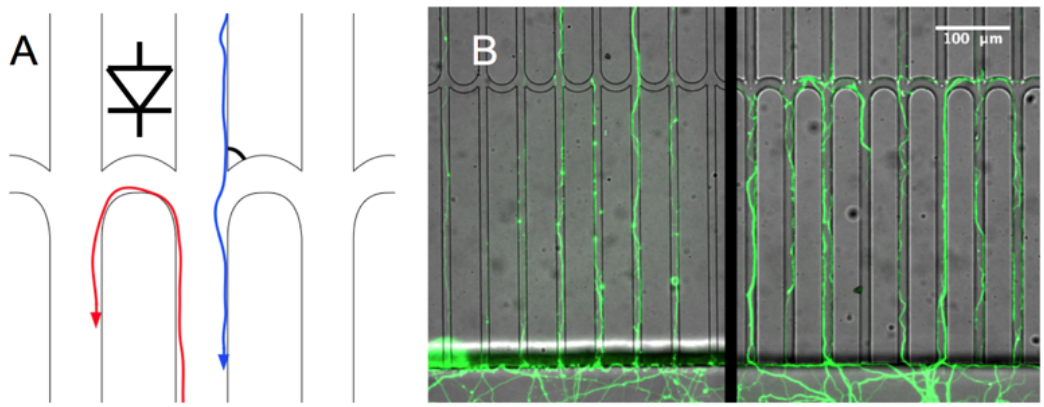


Figure 4.10. Experimental results on a new kind of axon diode. **A** In this design, axons are expected to experience the junction differently depending on their directions. The red axon grows along the edge which undergoes continuous changes of orientation and makes a U-turn : this is the blocking direction.
The blue axon continues straight as it can not fold around the corner, whose angle (in black) is smaller than the critical release angle : this is the forward direction. B- Experimentally, axons growing from the bottom to the top do preferentially go straight (left field) or make U-turns (right field) depending on the relative orientation of the junctions.

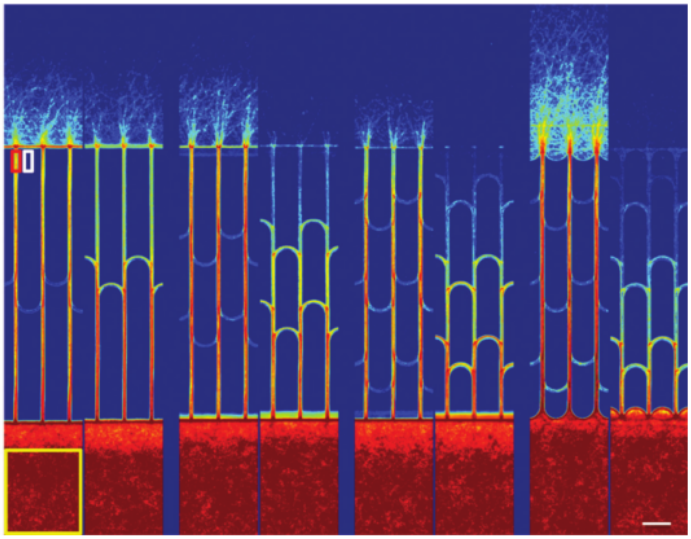


Figure 4.11. A mechanical diode for neuronal cultures Intrinsic direction selectivity in microfluidic chamber experiment. The bottom chamber is seeded with neurons, where they are left to grow. The affinity between Growth Cones and PDMS edges forces the axon to grow along the walls of the chamber, using this attitude, and the persistent direction of Growth Cone navigation it is possible to impose, mechanically, a preferred direction of outgrowing.
The shows the correlation between the number of archs in the diode structure and the probability of reach the top chamber. Image from [Renault et al., 2016]

4.4 From single GC to Neurite arborization, branching and competition

Why do neurons branch? How do the branches relate to each other? Is there a main branch? Is branching similar in 2 and 3dimensions? Does exist, and which is the optimal configuration for the dendritic tree?All this question are very deep into the comprehension of biological mechanism of neuronal arborization. This field has been highly investigated in the past and is nowadays, the understanding of neurite branching is far from be completed and while some biological mechanism for the generation of new growth cones has been successfully reduced to their minimal components the overall understanding of arborization process misses some relevant elements. To report the evidences and offer a complete review of the models used to mimic neuronal branching will require a separate work. Instead we will introduce the problem by the work of [Gallo, 2011] and [Cuntz et al., 2010]
. The objective of nervous system is to produce a functional information process system. The units of this system are the neurons and neuronal population, how to connect each unit to others in an efficient way is the challenge the evolution had to solve. The human brain cells solved the problem with the branching process: each neuron makes contacts with multiple targets through branching of its one axon, thereby reaching out to targets that are not in its direct trajectory, Fig. 4.12 presents an overview of the wiring problem

4.4.1 Evidences of branching *in vitro*

There are basically two ways of undergoes a branching process:

- **Bifurcation of the growth cone at the tip of the axon** during axon extension. Growth cone bifurcation generates two axon branches from the tip of the extending axon. This process contributes to axon guidance and to the development of the basic organizational scheme of the nervous system. However, branching through growth cone bifurcation is not the major mechanism that contributes to the sculpting of axons in their target fields.
- **Formation of axon branches from the axon shaft** behind the advancing growth cone. Fig. 4.12 **B**: Axon collateral branching – also referred to as interstitial branching – denotes the *de novo* formation of an axon branch from the axon shaft, independently from the growth cone present at the distalmost segment of the axon. The formation of axon collateral branches is widely regarded as the major mechanism used to establish axon arbors in synaptic target fields.

We are not focusing on the complex biological machinery to generate a new growth cone. The main idea, however, is the presence of a higher amount of Actin-F. Such excess can be caused by the growth cone stalling or the arrival of an *actin wave*. Once a protrusion is formed, the microtubules can invade the bulge and form the growth cone structure we have discussed before. Fig. 4.13 offers a schematic picture of the process.

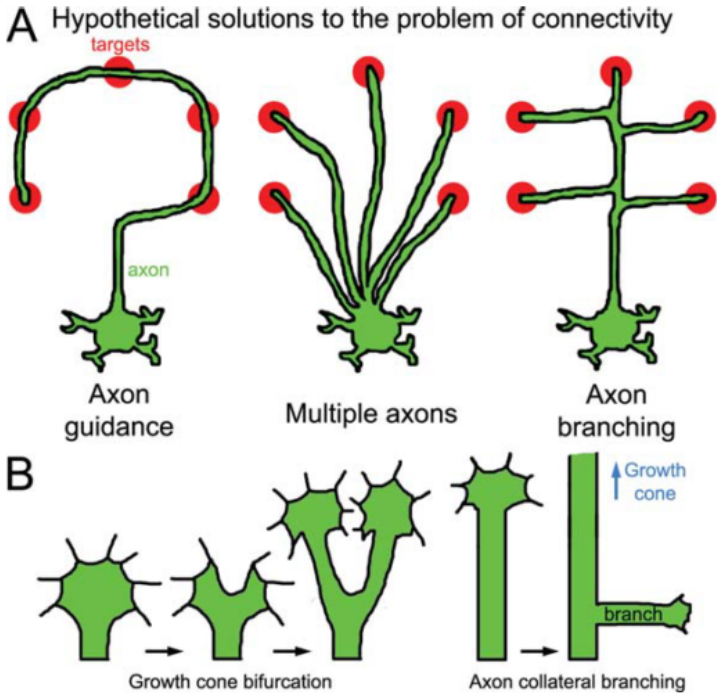


Figure 4.12. Modes of axon branching. (**A**) A single neuron could make contacts with multiple targets by: (i) projecting a single axon, guiding it alternately to each target; (ii) project multiple axons, each axon being guided to an individual target, or possibly to multiple targets as in (i); (iii) project a single axon which undergoes branching and each branch contacts one or more targets. Natural selection has continuously solved the problem of neuronal connectivity by adopting the strategy of axon branching (iii). (**B**) Growth cone bifurcation begins by the loss of protrusive activity at the leading edge while maintaining lateral activity. This results in the formation of two independently active growth cones giving rise to two axon branches from the tip of the axon. In contrast, axon collateral branches form *de novo* from the axon shaft after the growth cone has advanced past the site of branching. Collateral branches are initiated by the protrusion of filopodia or lamellipodia from the axon shaft. Image from [Gallo, 2011]

.

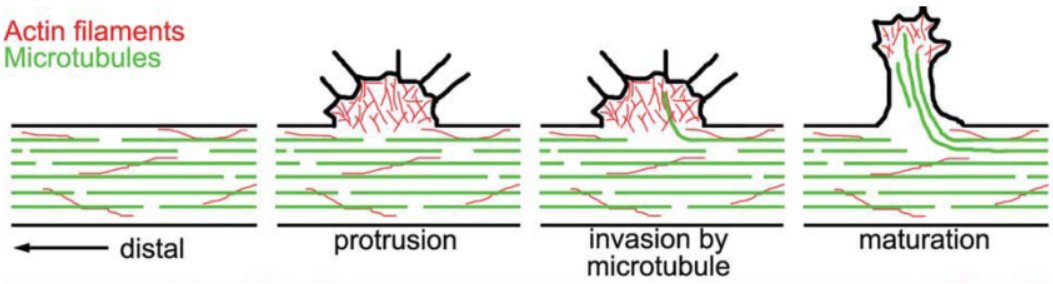


Figure 4.13. Rise of a collateral axon A basic sequence of the reorganization of the axonal cytoskeleton during branching. The first step involves the formation of actin-filament-based protrusions from the axon shaft. The protrusions are subsequently invaded by axonal microtubules, giving rise to a nascent branch. Maturation of the branch involves the entry of additional microtubules and cellular components into the nascent branch, e.g. organelles (not shown). [Gallo, 2011]

To conclude this short overview about the biology of outgrowing neuron, it is necessary to discuss briefly the problem of pruning. . In order to elongate the GC has to assemble tubulin in microtubule, but the building blocks it disposes are generally limited by the rate of synthesis in the soma [Goldberg, 2003]. Actually recent studies showed some protein are produced in the neurite itself, but anyhow the law of cytoplasm conservation by Ramon and Cajal has to be respected [Pestronk1995], establishing a competition among all extruding tips. Hence the neuron has to adopt strategies to optimize its growth towards effective directions: exuberant axonal projections generated during development must be differentially regulated so that beneficial branches are elongated while aberrant branches are eliminated [Oşan et al., 2011]. This large scale axon degeneration has been documented and studied in a variety of developmental systems. Ollustrative examples of both branching and branch elimination are shown in Fig. 4.14.

4.4.2 Competition

We have remarked the evolution shaped the genetics of neuron to fulfil their primary biological objective, create a network of information processing units. To reach the post-synaptic neuron is the primary task of outgrowing axons, we can consider pruning a method the neurons uses to optimize their neurites in this regard. While the navigation should be considered a semi-local behavior -both the chemotaxic gradient and the mechanical confinement are sensed by the GC itself, the neurite influences the Growh Cone with the microtubule rigidity, and, maybe, realising chemorepulsive signaling - branching and elongation are thought to involve the whole neurite or the neuron even. Indeed the law of cytoplasm conservation is showed to be - almost - enough to shape the morphology of a dendritic tree [Cuntz et al., 2010]. This constraint can be partially violated by the outgrowing dendrite, and these violations are the reasons of the broad zoology of dendritic morphology. A similar idea is sustained in the brilliant, thought old, review titled *How does an axon grow?*[Goldberg, 2003]. Here the relevance of the question -Does the axon requires

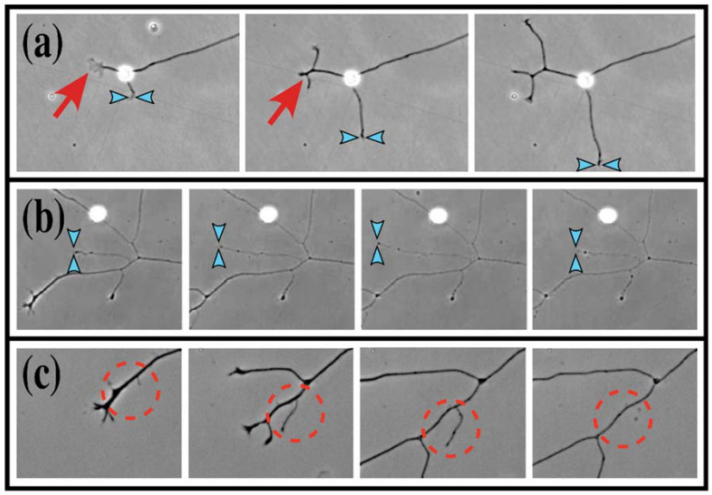


Figure 4.14. Time sequences showing branching and pruning of dissociated dorsal root ganglion neurites. (a) Branching (red arrow) and extension (blue arrowheads) of primary axons. (b) Extension and retraction (blue arrowheads) of neurite tip. (c) Tertiary branching and pruning (encircled). [Oşan et al., 2011]

building blocks from the soma to grow or it's self-sufficient?- is addressed and many experiments are reported to sustain the two hypothesis. Beyond the complexity of cellular biology simple questions can be drawn -the newly created cellular membrane is produced? Do the axon synthesize protein on its own or an diffusion/advective flux is received from the soma?- Thus when simulating the growth cone shaping the dendrites we need to respect these general constraints and broaden our look at the whole neuron. In the respect of these optimization principles we have settled from the point of view of the growth cones: they compete each other for the limited resource the neuron disposes. Some experiments were done to support these idea [Hjorth et al., 2014, Singh and Miller, 2005], and to find out which are the critical nutrients that affect microtubule synthesis, and therefore, elongation. Still is necessary to remark that the growth cone behavior is pretty variable and to make assertion on it requires a huge amount of measures to obtain a result statistically valid.

In the next chapter the straight observation of neuronal behaviour leaves the way to a biophysical attitude for modelling, towards a simplified behaviour that can be reproduced with a computer.

Chapter 5

Biophysical models for Growth Cone navigation and Branching

Contents

The evidences discussed in the previous chapter will now be presented in a biophysical fashion. The main difference is the perspective with which this discussion is addressed: the trade-off between exact description and mathematical feasibility is pushed and the experimental evidences give the way to reproducible models. In this chapter a bunch of already published models for GrowthCone navigation and neurite branching will be commented, these models were the basis for our modelling work. The chapter is divided into three sections, first the steering models will be discussed, then the GC elongation and eventually the branching process will be take on.

5.1 Modelling the growth

A full model of the neurite development and outgrowth, starting from the positioning of neurons in a Petri dish, would require a proper model of neurite initiation and differentiation. However, we can consider the axon distributed with uniform probability around the soma because of the lack of correlations between separated neurons and the isotropy of the space.

Once the first phase is over, the neurites, leaded by the growth cone, will be able to elongate and retract. The most relevant neurite is out of doubts the axon, whose elongation rate is way larger then the rest of dendrites, indeed the most of the articles concentrate on axon outgrowth. The study of dendrites and axon will be qualitatively the same: any evidence of a substantial difference between axon and

dendrites elongation mechanism is not reported [Suter and Miller, 2011]
In the absence of a definitive, biophysical, and solved model, the scientific literature offers plenty of particular models - often with a related *in vitro* experiment that tackle the many faces of the problem-. The following sections will review these modelsm highlighting pros and cons.

5.2 Steering models

It was very hard to find reviews or articles about the steering mechanism. This basic process seems to be relevant only for computational neuroscientist willing to reproduce the neuronal outgrowth *in silico*. The work we have kept in highest consideration during the implementation of the simulator is a previous project of Torben-Nielsen and Memelli [Memelli et al., 2013]. The authors suggested a environment agnostic model for neurite outgrowth: morphologies are generated by self-referential forces only.
In this model, branching probability follows a Galton–Watson process[1], and the geometry is determined by a drifted random walker, governed by internal forces. They modelled these forces, namely an inertial force based on membrane stiffness, a soma-oriented tropism, and a force of self-avoidance, as directional biases over a 3D random walk.
These authors didn't publish a detailed study of such forces, preferring to look at the morphologies and benchmark over them. In Fig. 5.1 the algorithm is described accurately.

5.3 Elongation models

In this section we consider three articles that most influenced our project. A preview of different models already used to describe the elongation of growth cones. None of the following articles claims to provide a full model of neuronal growth.

A mechanical GC Let's start from a completely mechanical model as the one proposed in [O'Toole et al., 2008]. The axon is considered as a viscoelastic system: a series of Burgers elements, springs and dash-pot, as presented in Fig. 5.2. This system was used to model towed growth of the axon, exerted with atomic force microscopy; the assumptions are that the axon reaction to force exertion enters in a steady state after few minutes and the system saturates its elastic properties. By these assumptions the elongation is due only to the dash-pot and is completely passive, induced by the pulling force; and it is possible to compute the elongation rate.
Assuming:

1. In the dashpot the resulting speed depends linearly by the exerted force, $v = F/G$

[1] A standard branching process introduced by Francis Galton's for a statistical investigation in the extinction of family names.

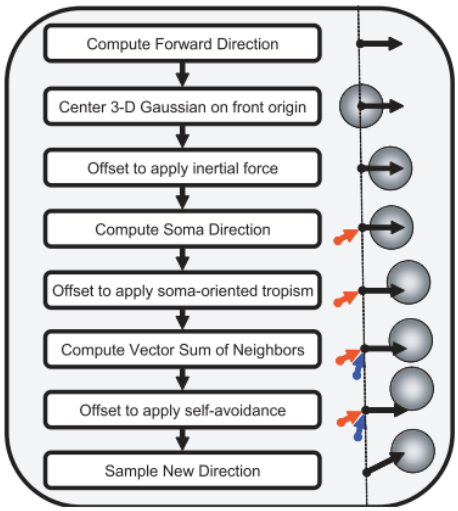


Figure 5.1. Memelli and Torben-Nielsen steering algorithm The algorithm used in [Memelli et al., 2013] is a superposition of directional vectors computed through the parameters of the different self-referential forces.
The first parameter, the inertial force can be computed on local information only, while second and third forces require the position of the soma and the neighbour cones. The former can be easily stored in memory but the latter require full space perception at each step of the simulation.

2. The neurite contributes to the elongation with all the segments and not only by the GC tip

3. Some segments of the neurite are anchored to the substrate with a dash-pot of constant η

<++> then a simple system of differential equations can be solved and the following equations state:

$$L(t) = \sqrt{(G/\eta)}\sinh(\beta \exp(\frac{F_0}{G}t)) \tag{5.1}$$

The model forecasts a stable rate of elongation for each element of the neurite, which ends up in a non linear growth rate at the neurite tip. This model is inconsistent with the growth cones we want to model, which are not pulled but instead grow spontaneously. However the model is quite simple and solvable analytically.

MAP2 diffusion model A completely different model is proposed in [Hely, 2001]. Here the author consider a compartmental neurite and the diffusion of a certain protein on it. Such protein is the *Map2* which has a relevant impact on microtubules fasciculation and stability. The protein can phosphorylate or dephosphorylate [2]. The

[2] In biology, phosphorylation and its counterpart, dephosphorylation, are critical for many cellular processes. Phosphorylation is especially important for protein function as this modification activates

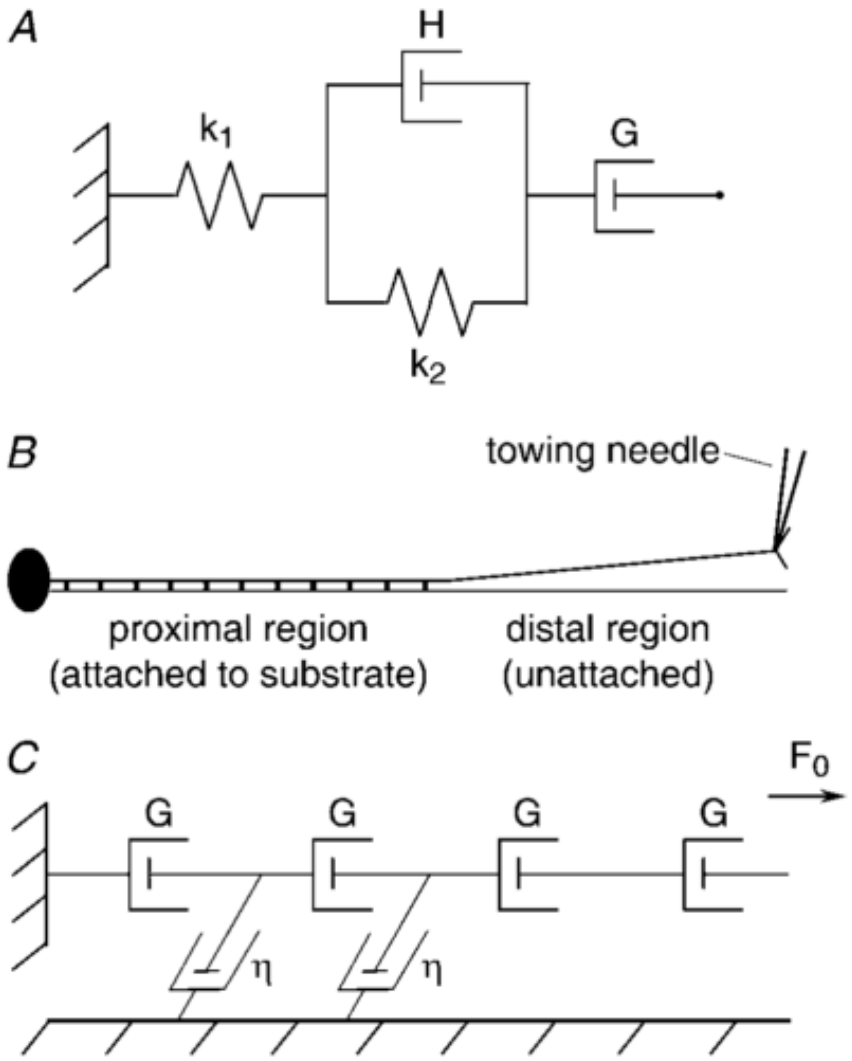


Figure 5.2. Model of a towed neurite as a series of dashpots. A Consider the axon as a series of Burgers elements. Each element consists of two elastic elements and a free dashpot (with constant G), which simulates towed growth. **B** Diagram of a neurite during towing. The distal region of the neurite is free of the substrate, whereas numerous adhesions in the proximal region cause the neurite to remain firmly fixed. **C** Under constant tension (F_0), the behavior of a Burgers element is dominated by its free dash-pot.

The model treats a neurite under constant tension as a series of dashpots. Attachments to the substrate are represented as friction dashpots (constant h). Tension is constant in the distal region but is dissipated by adhesions in the proximal region. [O'Toole et al., 2008]

activation (phosphorylation) of MAP2 enzymes reduce the probability of branching and enhance the elongation rate. In the hypothesis

1. the phosphorylation is regulated by the amount of Calcium in the compartment.
2. the MAP2 protein is produced only in the soma.
3. the Calcium influx depens only by neurite diameter.

the author sketches a schema of 4 linear differential equations, reported in Fig. 5.3. Once the MAP2 dynamics is integrated in a compartmental neuron, the elongation rate E and branching probability P can be computed by the following equations:

$$E = E^0 k_E MAP2_b \frac{MAP2_b}{MAP2_p} \tag{5.2}$$

$$P_{br} = P_{br}^0 k_B MAP2_p \frac{MAP2_p}{MAP2_b} \tag{5.3}$$

This model is very interesting for the simulator project since it is quite omni-comprehensive and accounts for the elongation rate as well for the branching probability. Nevertheless the solution of the linear system requires a compartmentalization of the neuron and the compartments number grows exponentially with the branches in the neurite. We have maintained the general idea of elongation as a byproduct of local factors, the Calcium influx, and soma generated proteins, using respectively neurite diameter and competition as proxies. The link between elongation and branching is also maintained in the model implemented in NetGrowth.

Growth, collapse and stalling Collapse and stalling are introduced in a fully biophysical fashion in [Recho et al., 2016]. The author describes the neurite as a viscoelastic system, imposing and solving set of conservation equations - conservation of mass, momentum and rheological assumptions. The neurite undergoes the forces exerted by microtubule invasion of growth cone and F-actin meshwork contractility. This model, which is based on only biological and measurable parameters - actin and tubulin meshwork friction parameters, polymerisation rate, etc. . . - predict success the existence of three stages of neurite outgrowing. These three phases, growth, collapse and stalling, are regulated by the load at the end of the neurite, generally a pulling force. Whether the growth cone is free to elongate the pulling force is the actomyosin molecular motors that pull on the focal adhesion created by the growth cone on the substrate. Let's define some variables: $a = \frac{\zeta_c}{\zeta_\mu}$ is the ratio between the friction of microtubule with the neurite cortex and the friction of cortex with the substrate; hence the Growth Cone exerted force is $Q_{gc} = \sqrt{a} v_p$, where v_p is the polymerisation rate of G-Actin in F-Actin. The derived equations are quite complicate but can be summarized in three regimes:

- **Retraction**, whether Q_{gc} is less then the stress exerted by the microtubule turnover Q_μ
- **Stalling**, if Q_{gc} is comparable with the friction of the neurite cortex Q_s and microtubule turnover, and the total length is predicted in this case.

(or deactivates) almost half of the enzymes, thereby regulating their function

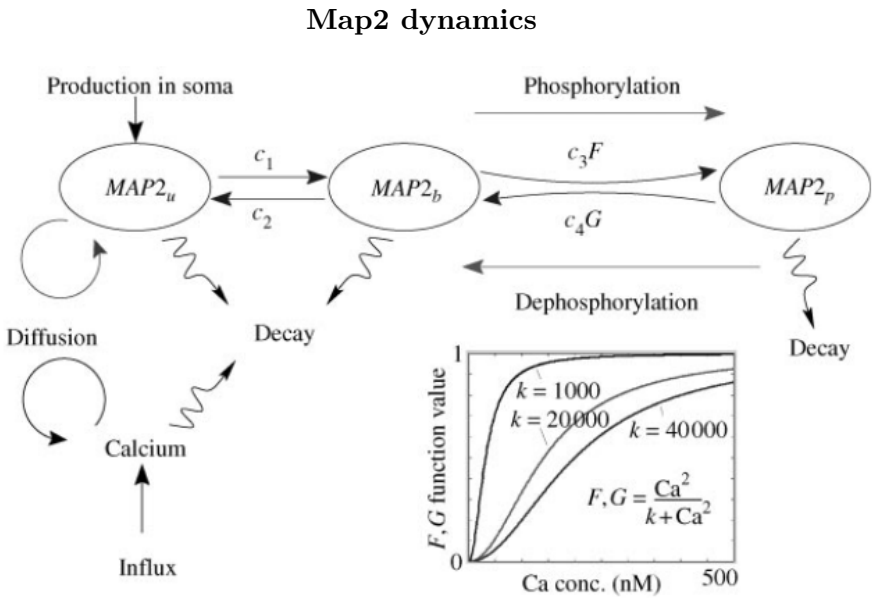


Figure 5.3. The cell dynamics simulated in the branching model. Diffusible $MAP2_u$ is produced in the soma. First it is converted into the microtubule-bound $MAP2_b$ complex and then phosphorylated into $MAP2_p$. Calcium enters across the membrane and diffuses throughout the dendrite. The calcium-dependent functions F and G control the on/off rate of phosphorylation. The shape of these functions is shown for various values of the steepness parameter k used in the simulations.

- **Elongation** with velocity

$$V = \frac{Q_{gc} - Q_s}{2\sqrt{(a)} + B} \tag{5.4}$$

  where B comprehend parameters we have not introduced, generally linked with neurite width and actin viscosity.

This model is very well done, both for the deepness of computation, where each assumption is justified with experimental evidence, both for the overall simplicity of the predicted results. To conclude its study the author has performed pharmacological studies, in very good agreement with the predictions. Fig. 5.4 summarizes the resultss We have replicated the existence of this three thresholds in NetGrowth, but there is not an equivalent model for the stresses and frictions exerted by the neurite components. We hope to implement this model and verify the reproducibility of Recho's results in a future model of NetGrowth.

5.4 A mathematical picture of neurite branching

Let's now tackle the complex problems of GC generation. The initiation of branches -and the consequent arborization- shapes the neurites, their computational properties and the number of synaptic connections. It is fundamental to reproduce the neurite tree adequately.

Several modelling approaches have been used in the *in silico* synthesis of dendritic trees. These can be characterized either as Growth Models or Reconstruction Models. Growth Models are based on principles of dendritic development, using rules of outgrowth associated with dynamic growth-cone behaviour and microtubule-mediated neurite elongation. In contrast, Reconstruction Models use an algorithm based on a canonical set of elementary properties which are originally derived from the characterization of existing dendritic structures [Polavaram et al., 2014]. Note that Reconstruction Modeling is a purely descriptive approach which uses minimal rules to *synthesize* topologically-realistic neurons.
In contrast, GrowthModeling adopts an exploratory approach by using biological rules of development and observations of the outgrowth process to explain or predict variations in full-grown arbor structures.
These two modes of branching give rise to different geometrical patterns of axon branching and generate different aspects of neuronal circuitry. For the NetGrowth simulator the choice was obligatory, a full generative model was due to mimic the wall-GC interactions.

BEST model Altough the biological mechanisms and the relevant parameters were not fully understood, Dutch researchers Van Pelt and Van Oyen gave a great contribution to the understanding of outgrowing neurites. They have published several articles [van Ooyen et al., 2014, Van Pelt and Uylings, 2003, Van Pelt et al., 1997] to face this problem with a simple and effective mathematical model. In [Van Pelt and Uylings, 2003] they propose the **BEST** model: the branching mechanism is GC splitting only and the parameters are completely phenomenological. Given the number of segments n, the competition parameter among GCs E and a baseline distribution for the branching rate $D(t)$ [3], the growth cone branches with probability:

$$p_i(t|n(t)) = \mathcal{N} n^{-E} 2^{-\gamma_i S} D(t) \tag{5.5}$$

with the normalization factor (5.6)

$$\mathcal{N} = \sum^{n(t)} 2^{\gamma_i S} \tag{5.7}$$

It is possible to show, for an exponential decaying baseline, the number of branching events is limited:

$$n_B(t) = n_\infty^{1-e^{-t/T}} \quad \text{for E>0} \tag{5.8}$$

Here γ_i is the centrifugal order[4] and the parameter S is a symmetry parameter a sketch of such process is offered in Fig. 5.5.

BEST model is an artificial solution, where parameters have to be fitted with experiments and can variate slightly from a neuron type to another. The converging integral $\int_0^{infty} dtD(t) = AT$ can be associated to the average number of branching events B, while the time scale is fixed by the characteristic time T. The parameters

[3] The branching baseline is necessary to give a temporal frame to the BEST model: the first implementation [Van Pelt and Uylings, 2003], used discrete distribution and constant time binning, without any relation to time. Such a model whether reproduced the right statistical measures were wrong in terms of branching time.

[4] Centrifugal order is the number of nodes separating the node i from the root

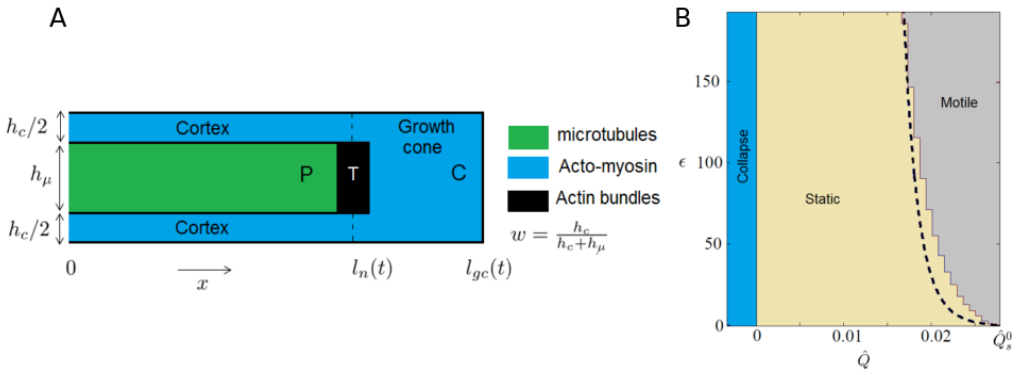


Figure 5.4. Growth, collapse and stalling from biophysical quantities The model proposed by Recho, and summarized by Eq. 5.4, starts from a simple depiction of the growth cone internal forces, the one proposed in **A** and develop a set of viscoelastic equations. After some evidence-based assumptions the model is simplified and a two thresholds are derivate: a stalling threshold and a retraction one. The first depend by the microtubule turnover rate, while the latter by the contractile load of actin cortical network. Picture **B** show the growth, stalling and retraction regions varying the pulling load, i.e. growth cone force in our case, and the parameter ϵ, this is the ratio of acto-myosin molucular motor over the microtubule network viscosity [Recho et al., 2016].

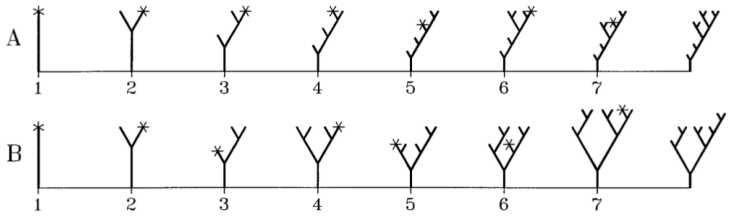


Figure 5.5. VanPelt BEST model The picture represents the time evolution of two Van Pelt trees. The constant baseline function – namely $D(t) = B/N$ – is the average number of branch events. The sketch shows two different sequences, where the same stochastic model generates two different trees. The time component $D(t)$ is introduced to fit real neuron measurements. Switching from a discrete-time to a continuous-time probability function impacts the theory and prevents the number of branching events from divergence thanks to the exponential decay of the function $D(t)$ [Van Pelt and Uylings, 1999]

.

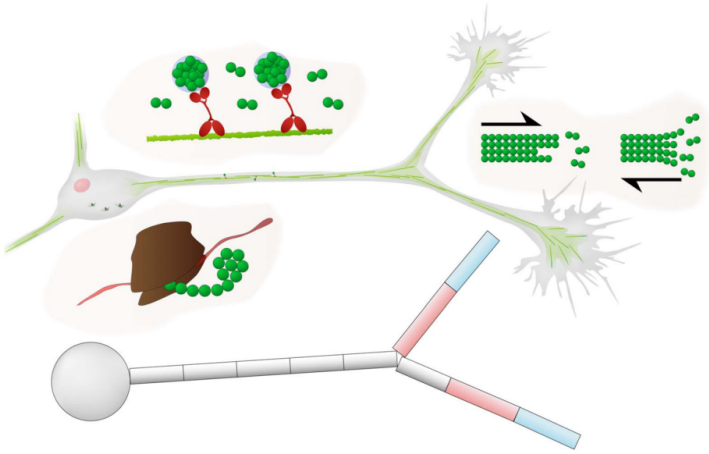


Figure 5.6. Tubulin dynamics in the model. Tubulin molecules (green spheres) are produced in the soma by biological neurons via translation of mRNA on ribosomes (the brown structure). Tubulin is then transported by diffusion and active transport. In biological neurons, the microtubule bundles (the light green fibers) act as railway tracks on which the tubulin molecules are bound via motor proteins (the red molecules). Tubulin is transported to the growth cones at the tip of the neurites. At the growth cone, tubulin is either integrated or polymerized into the microtubule cytoskeleton (long polymers of tubulin dimers – green fibers), which elongates the neurite. When the microtubule depolymerizes, the neurite retracts and tubulin becomes free again.

optimization is performed considering the mean and the variance of measured real neurons, as those retrievable in the NetMorpho database.

5.4.1 Many GCs compete with each other

The pruning introduces the problem of competition for the growing neurites. Which branch is going to collapse? Is the optimization process of the tree structure local or global? Is possible to model arborization as a competition between growth cones? The microbiology of neurotransmitters that trigger this process is a complex subject, but we can give a general idea of it by a simple mathematical model. In fact, the competitive dynamics introduced by [Hjorth et al., 2014] offers a simple perspective over the whole phenomenon. Some elements – required by the GC to synthesize microtubules and then elongate – are built into the soma, thereby needing to be transported to the very end of neurite's tree in order to be available for the growing GC. This transportation can be diffusive or active – i.e. done by molecular motors – and the two cases lead to very different results. This simple idea was implemented with a compartmental simulator, where each segment of the neurite stores or consumes some of the limited material. In the cited work, the referred protein is tubulin. Fig. 5.6 presents a scheme of its production and polymerization. During retraction tubulin is depolymerizated.

The equations of the model can be condensed in the following one that represents

the evolution of the amount of tubulin Q_i in a certain compartment:

$$\frac{dQ}{dt} = Df(Q, A, V) + \nu g(Q, A, V) - bQ - X\frac{dL}{dt} \tag{5.9}$$

the equations govern the evolution of the tubulin amount $Q_i$in a certain compartment, the r.h.s first term is the diffusive one and is moderated by the diffusion coefficient D, the second term represents the active transport regulated by ν, b is the tubulin corruption rate and, eventually, the last term $X\frac{dL}{dt}$ accounts for the consumption of tubulin in the growth cones. The tubulin diffusion ODE is associated by an elongation ODE for each terminal segment:

$$\frac{dL}{dt} = pQ - q \tag{5.10}$$

The elongation rate is defined as the result of the polymerisation rate p times the tubulin Q in the leave, minus the depolymerisation rate q. An experiment with this model is detailed in Fig. 5.7.

The model is quite simple but requires the integration of the whole ODE step by step and is not very suitable for an efficient simulator. The competition model for critical resource we will introduce later is largely inspired by this model.

5.4.2 Optimal wiring and scaling laws

Competition among growth cones can be by another perspective, like the one proposed by Cuntz and collaborators[Cuntz et al., 2010, Cuntz et al., 2012]. Instead of generate the neurite step by step through growth cone migration, the authors of TREESToolbox [5] have proposed an optimization-driven approach for neurite reconstruction: the algorithm builds tree structures which minimize the total amount of wiring and the path from the root to all points on the tree. On the basis of Ramon y Cajal cytoplasm conservation law, they propose three more detailed principles:

- By adjusting the balance between the two wiring costs -total dendrite length and distance from the soma-, a dendrite can efficiently set its electrotonic compartmentalization, a quantity attributable to computation.
- The density profile of the spanning field in which a dendrite grows determines its shape dramatically.
- A few weaker constraints such as the suppression of multifurcations, the addition of spatial jitter or the sequential growth of sub-regions of a dendrite are helpful for reproducing the dendritic branching patterns of particular preparations.

Unfortunately there is not any direct application of these additional constraints in a generative simulator. Nonetheless they might shed light on further functional, computational, developmental or network determinants for certain dendritic structures, we have tried to take in to account these constraints in the competitive model,

[5] `http://treestoolbox.org`TREESToolbox

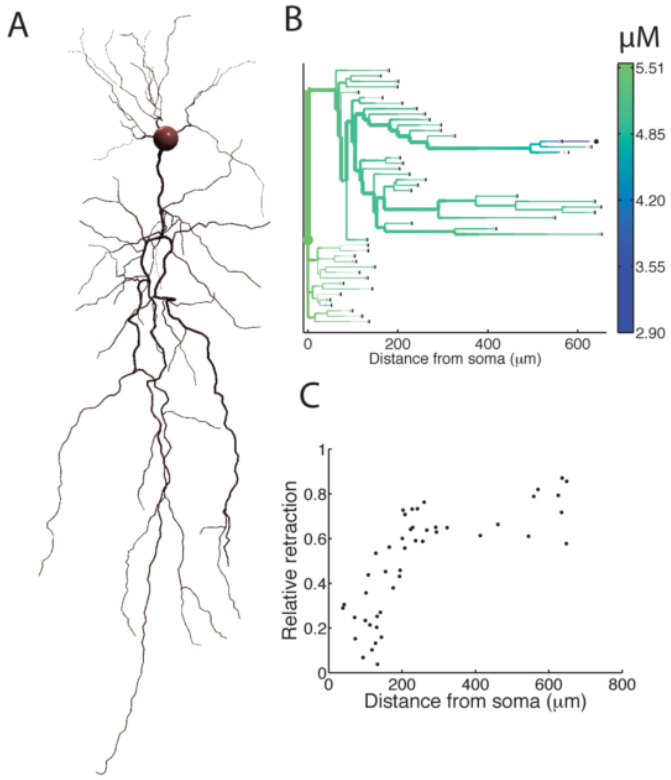


Figure 5.7. Competition between neurite branches in a complex morphology. (A) Example morphology of a reconstructed pyramidal neuron with apical and basal dendrites. (B) In the control case, starting from the reconstructed morphology, the neuron was allowed to grow out for 10 hours in the model. The simulation was then repeated with the same initial conditions, but with an increased polymerization rate for one of the growth cones. The dendritic morphology obtained in this last simulation is represented by a dendrogram, coloured according to the tubulin concentration in the branches. The gray vertical lines at the terminal segments indicates the starting morphology, and the black vertical lines show the neurite's length after 10 hours – in the control case. The black dot marks the growth cone with increased polymerization rate. (C) The competition between branches increases with increasing path distance to the soma. The graph shows the total retraction of all neurites divided by the growth of the modified growth cone, as a function of the path length between the modified growth cone and the soma. [Hjorth et al., 2014]

e.g. setting an explicit attenuation of the critical resource received when the -user defined- total dendritic length is exceed. Most of all the simple principles presented in this study can be used to efficiently edit, visualize, and analyze neuronal trees and will be used to validate the morphology generated.
On this purpose the result accomplished in [Cuntz et al., 2012] is even more relevant: although the wide diversity of dendritic trees, they have developed a general quantitative theory relating the total length of dendritic wiring to the number of branch points and synapses. The optimal wiring presented above predicts a 2/3 power law between these measures and they show the theory is consistent with data from a wide variety of neurons, across many different species.

5.5 Gateway to stochastic models

The discussed models are deterministic as a consequence of solving the macroscopic equations of mechanical or chemical processes. However the simulation of such complex events cannot be fully deterministic since the phenomenon undergoes non-equilibrium statistical mechanics. A weak argumentation is to consider the many undergoing processes we are neglecting, proteins diffusion, synthesis, and the chain effect on the growth itself; the branching itself is depicted by a system with a variable number of ODEs (equation are doubled after first branch). This is enough to expect a deeply stochastic process and to introduce stochastic elements into the simulation. The Netgrowth simulator models the outgrowth dynamics of axons and dendrites with stochastic rules, at many levels. The steering is regulated by a random walk, the branching is mediated by a stochastic model and the neuron position is drawn by a spatially uniform distribution.

Chapter 6

Implemented models in NetGrowth for neuronal outgrowth in constrained environments

Contents

Most of researchers focus into study a particular aspect of growth cones migration, e.g. the elongation rate neglecting the steering mechanisms and vice-versa. Although this method is scientifically correct, complex simulators, as NetGrowth, requires to mix these models to reproduce a plausible neurite outgrowth. Furthermore some of the algorithms and mechanism discussed are computationally expensive, e.g. the compartmental model which is widely used to describe the growth cone elongation[1].

[1] It requires the integration of an increasing set of ODEs: the integration time goes at least with the number of segments in the tree, which is exponentially growing in time

Hence the ideas are kept and the algorithms rearranged to fit with NetGrowth scopes.

Outline of the chapter In this chapter we are going to describe in details the models used for the NetGrowth simulator, which I personally developed and tested. We will tackle each problem separately: the steering process, the interaction with the environment and the elongation rate will be discussed. Some of the largest digression are moved in the Appendixes, to keep the text linear and coherent. The chapter is divided in two sections, in the former I will look at single growth cone migration: the way they interact with the environment and move towards their biological targets: the isotropic model is discussed previously and then the environment-aware one.
In the latter section I will introduce the whole neurite: first a stochastic equation for neurite branching is presented, later the competition among growth cones of same neurite is studied with a system of two SDE.

6.1 Growth Cone migration: Steering as a random walk in 2D

The easiest way an agent can explore the space is carrying out a 2D Random Walker, a Markovian process largely used in physics. We want to verify if the RW reproduces adequately the navigation of the Growth Cone on the substrate. The hypothesis is the biological procedures to steer and explore the environment can be mimicked with a proper algorithm. Hence the algorithm is expected to simulate the migration of a Growth Cone in a physical constrained environment.
In the first section I will introduce the problem and offer a general picture of the GC dynamics in the NetGrowth simulator. In the second section the relevant properties required to the algorithm will be presented, some algorithms will be outlined and the results discussed. Eventually some real neurons will be analysed and compared to the simulator. This short section focused a relevant part of the project: the navigation is a fundamental aspect of the whole simulation. Some tools, theoretical and computational were used to characterize the algorithms, they are discussed in AppendixA

6.1.1 Modelling the elongation with a random walk

The random walk is very easy to simulate and code, to associate it to a Growth Cone is straight-forward: the active walker is the GC while the previous location are joined up to form the neurite. This simple fact implies the walker cannot cross itself constantly and the local aspect of the trajectory is relevant. Indeed in physics or finance the random process are often 1D and in general the history is not relevant. A larger category of walker are the self-avoiding paths, but they have not been used in the present work for they are computationally expensive. Some classical algorithms, whose analytic feasibility make interesting, like the Ornstein-Uhlenbeck process, were set aside for they are definitely not self avoiding and the trajectory cannot resemble anything biologically consistent
The Growth Cone moves on the xy plane but it's evident the random walk is actually

on the angle. For a subset of algorithms the step can be fixed and the whole process depends by the distribution of new angles. In this case a simple consideration is that the GC goes as much straight as the angle changes slowly. Such a naive observation hides the complexity of the relation between the mean square displacement in 2D and the one regarding the angle variable.
Hence the coordinates system used is the polar system with the origin centred on the soma, where the RW starts, and the angle initial condition set to zero:

$$\vec{r}_n - \vec{r}_{n-1} = \hat{x} \cdot \rho \cos\theta_n + \hat{y} \cdot \rho \sin\theta_n \tag{6.1}$$

The step is generally not fixed, but the speed is. Generally speaking to compute the evolution only the stochastic variable θ is drawn.

6.1.2 Geometrical requirements for the RW algorithm

Outgrowing GCs are constrained to go straight-forward from the rigidity of micro-tubules. This simple argument exclude the possibility of drawing the angle from an isotropic distributions, as the uniform distribution over 2π. The next step direction has to be influenced by the previous in the angles variable space, the process, where coordinates have a temporal correlation, is also known as Brownian motion. We can set whichever functional form to determine the walker stochastic evolution but we started from the Gaussian. Such idea relays on some general results from cultured neurons' experiments [Renault, 2015]; and a micro-physic explanation of such behaviour is based on simple statistical mechanic system fluctuating around the minimum energy. Let's consider the micro-tubules stiffness k_{mt} and the energy required to bend the neurite with a certain angle, looking at the steering as a fluctuation from the equilibrium:

$$\Delta E = \int_{\theta_0}^{\Delta\theta} k(\theta - \theta_0) d\theta = k\Delta\theta^2 \tag{6.2}$$

$$P(\Delta\theta) \sim e^{-\Delta E \beta_{eff}} \sim \exp(-\frac{\Delta\theta^2}{2}\beta_{eff}) \tag{6.3}$$

Where β_{eff} collects all the system chemical and thermal details and is interpreted as the variance σ. Other dynamics, with a more robust theoretical background, as those leading to Levy-flights or power-law distributions are presented in [Metzler and Klafter, 2004]. To study the Gaussian model and check its results is an useful work, at least to show its limits.
Let's recall briefly the growth cone we have described so far, the factors we want to take into account are the following:

- **the stiffness of micro-tubules** Neurite's microtubule core prevents the system from abrupt turning. [Martin, 2013].
- **the soma-tropism of neurite**: Neuron's biological objective is to reach out other neurons. The neuronal targets are reached by the growth cones which undergo self-referential forces preventing from self-rolling and back-turning. [Memelli et al., 2013]

- **Space isotropy**: The growth cones haven't any spatial preference in an isotropic environment and everything is symmetric respect the extrusion angle.
- **Ballistic assumption**: The measures performed on real neurons show ballistic behaviour, the GC directs straight-forward away from the soma.

Furthermore we have to consider the requirements set by the simulator itself:

- The computational cost must not increase with the distance from the soma.
- The algorithm has to be expanded to an environment interaction model.

Furthermore we define a physical property, the persistence length l_p.

Correlation and Persistence Length The correlation in 2D can be defined through a vectorial picture of the process: Let's $\vec{b_i}$ the element of the trajectory γ which the segment $i, j, \ldots, n$ belong to. Let's each step has the same length b. The correlation between the segments is:

$$C(i,j) = \frac{\langle \vec{b_i}\vec{b_j} \rangle}{b^2} = \langle \cos(\Delta\theta_{i,j}) \rangle \tag{6.4}$$

Assuming the process in a stationary state, the correlation function will depend only by the distance $s = i - j$, whether this is not the case we set the site i constant to zero and j stays the only variable. Hence, developing the function around $\Delta\theta \sim 0$:

$$C(s) = \langle \cos(\Delta\theta_s) \rangle = \sum_0^\infty \frac{(-1)^n}{(2n)!} \langle (\Delta\theta_s)^{2n} \rangle$$

while $\Delta\theta_s$ remains close to zero, the expansion can be reduced to the quadratic term.

$$C(s) = 1 - \frac{1}{2}\langle (\Delta\theta_s)^2 \rangle \tag{6.5}$$

In this picture of the process the correlation is still an abstract quantity, it does depends by the time interval and misses a characteristic length. It is way more useful to define a characteristic length that can be obtained measuring mature neurons, an experimental measure of microtubules persistence length is showed in Fig. 6.1. The persistence length should be invariant on coarse graining of the trajectory, while the correlation between two segments distant s time intervals changes when the time step is augmented or reduced.
Such characteristic length is a general property of local processes in physics, while it disappears in phase transition and long range correlations: the functional form of correlation depends by the recursive equation, it is exponential decaying for short range phenomena and power law for long range. The algorithms we will use belongs to the first class and thereafter we can define the persistence length, regarding to real neurons this cannot be said, because some process, i.e. self-avoiding walks, lacks a characteristic length and the correlation decay slower. Whether the real neurons should belong to the latter category different algorithms will be implemented. To characterize the neurite properties the persistence length is a precise tool. It is expected to be a macroscopic property of the system, the average length of a straight

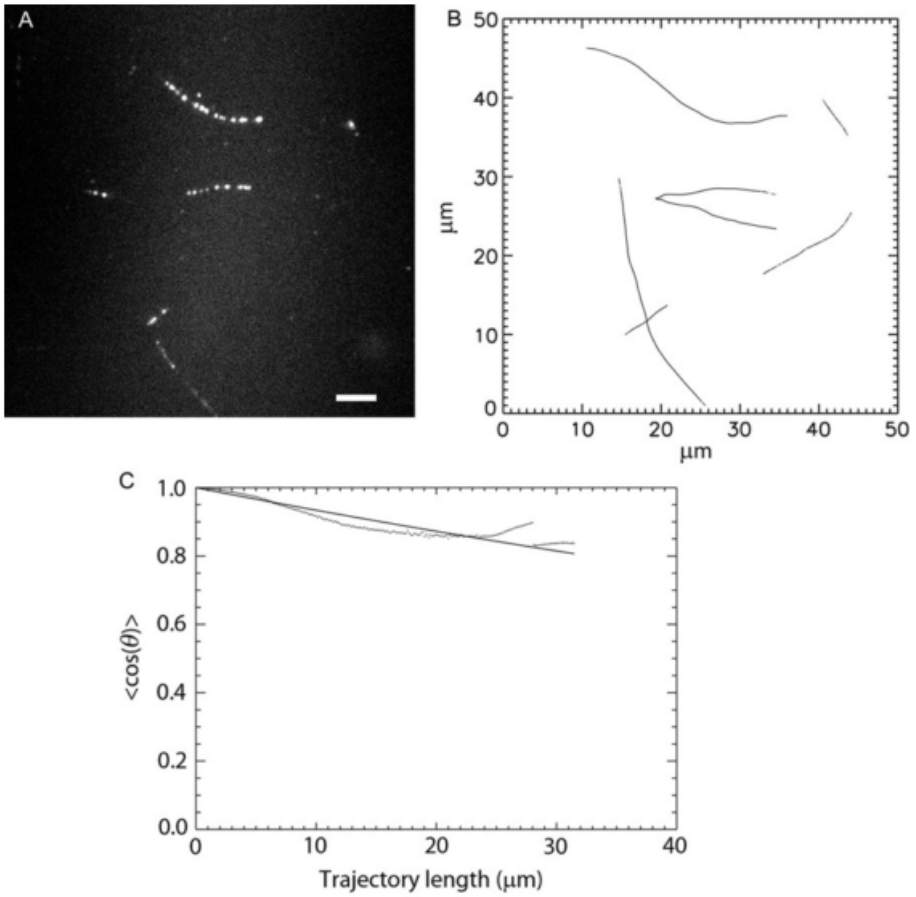


Figure 6.1. Persistence length of dissociated microtubules. The persistence length can be calculated from mature cultured neurons *in vitro*. Measures the persistence length of dissociated microtubules can help to forecast the relation between the neurite diameter and its stiffness. This article is particularly relevant because implement a method to measure the persistence length for fixed microtubules. The pictures report results from the experiment documented in [Martin, 2013]. (A) TIRF image of microtubules sparsely decorated with fluorescent GTP analogs. Scale bar is 5 mm. (B) Microtubule trajectories from (A). (C) Average of $\cos(\Delta\theta_s)$ as a function of path length, s. The data shown are an average across all trajectories in (B). Fewer independent data for long trajectory lengths lead to statistical fluctuations. *The persistence length for these microtubules is 150 μm*

segment originating in the soma. The persistence length depends by the parameters of the problem and is a property of the physical system, which has a well defined space-scale. The persistence length can be obtained placing the function $C(i)$ in the real space. Assuming the system varies continuously with a function $\bar{x}(t)$, we have some points of the curve, the resolution doesn't affect the problem since the scale is fixed. The variable s is the length of curvilinear path, that is the length of the dendrite itself. The variable s is measured in μm and the persistence length is defined as:$l_p = min\{s : C(s) < e^{-1}\}$

$$C(l) = \exp(-\frac{l}{l_p}) \tag{6.6}$$

6.1.3 Choosing the appropriate algorithm, pros and cons of different descriptions

Bearing in mind the properties outlined in the previous section we will describe some algorithms and analyse their weakness and strengths. A detailed characterization for the algorithm is offered in Appendix A here it is limited to maintain the unity of the manuscript. There are two main classes: Those implementing a stochastic equation for the angle variable; and those presenting an unified dynamics, with the elongation speed influencing the directional dynamics directly. Some algorithms are compared in 6.9.
The first algorithm on the stage is the Correlated Random Walk, CRW. The dynamics is simple and is governed only by the following equation, r represents a correlated Gaussian variable, with unity variance and zero mean:

$$r_n = \beta \cdot r_{n-1} + \sqrt{1-\beta^2} \cdot \xi_{n-1} \tag{6.7}$$

$$\theta_{n+1} = \theta_0 + (r_n)\sqrt{\sigma} \tag{6.8}$$

$$r_0 = \xi_0 \tag{6.9}$$

Where ξ_n is a Gaussian distribution with mean zero and variance one. The angle θ_0 is set and the neurite produce a river like path around its direction. The persistence length of CRW is set by the parameter β for the angle variable, while the neurite does not experience turns. This algorithm produces a naive behaviour, the GC outgrowth and keeps going straight for all the growth process, whether there is a resemblance for some neurons' sample it is not usable in the NetGrowth simulator. Indeed it depends by the initial conditions, whether this is fine for free space neurons, those constrained by the environment will present bizarre behaviours.
A plausible correction for the CRW is to set dynamically the angle θ_0, avoiding the crucial problem described above. The new algorithm presents a short term memory that variates the direction, the equation for the Memory CRW follows, while a

descriptive picture is offered in Fig. 6.2

$$\begin{aligned} \bar{\theta}_n &= \frac{\sum_{i=0}^{n} \alpha^{n-i}\theta_i}{\sum_{i=0}^{n} \alpha^{n-i}} \\ r_n &= \beta \cdot r_{n-1} + \sqrt{1-\beta^2} \cdot \xi_{n-1} \\ \\ \theta_{n+1} &= \bar{\theta}_n + (r_n)\sqrt{\sigma} \end{aligned} \tag{6.10}$$

Initialized with:

$$\begin{aligned} r_0 &= \xi_0 \\ \bar{\theta}_0 &= \theta_0 \end{aligned}$$

The term 6.1.3 averages over the trajectory: takes into account previous directions and allows the random walker to variate its original direction. Memory acts as a rotation and doesn't take part to the stochastic computation. It is parametrized with α and dominates the persistence length. Parameters α, β and σ cannot be chosen in the whole space or pathologic behaviours will appear: i.e. the correlated Gaussian cannot be larger of memory constraint or the walker will rotate on itself. They are not orthogonal, as it will be shown later, and it's not possible to map one-to-one to stiffness, soma-tropism, etc. This model fits better for the NetGrowth scopes and it has been implemented in the simulator, whether it cannot be related easily to the biology of the microtubules its worst drawback is another. The algorithm presented leads to very complex discrete differential equations, which cannot be solved (the detailed computation is in the Appendix A). Hence the persistence length cannot be obtained analytically by the parameters. This won't be a problem if a numerical estimation could be performed, but the algorithm produces 2D processes whose persistence length changes in a super-exponential fashion in respect to model parameter; it becomes very hard to fit via the three different parameters, as illustrated in Fig. 6.4. This annoying properties, added to its non linear dependence by the resolution of time step (indeed for higher resolution the number of turns is reduced abruptly) which cannot be correct with multiplicative factors, makes the MCRW very hard to pass the "reproducibility" test. Whether it can produce interesting shapes as showed in Fig. 6.3, it is not suited to predict neuronal network from biological data. The impossibility to produce sensible behaviours with such simple algorithms constrained us to rethink the whole stochastic process. The first interest remains to produce a persistence length l_p that can be set by the user *before* the simulation, maybe from data obtained in neurobiology laboratories. The complexity showed by the MCRW requires to make a step backwards toward a more predictable algorithm: The simplest case we can imagine is the Diffusion RW, as the name suggest the angle variable changes with a simple diffusive stochastic equation, and is governed only by the variance parameter σ:

$$\theta_{n+1} = \theta_0 + \xi_n\sqrt{\sigma} \tag{6.11}$$

In this case the angle shows a completely diffusive behaviour and, as it is showed in the appendix, it leads to a double scale dynamics for the GC in the xy plane.

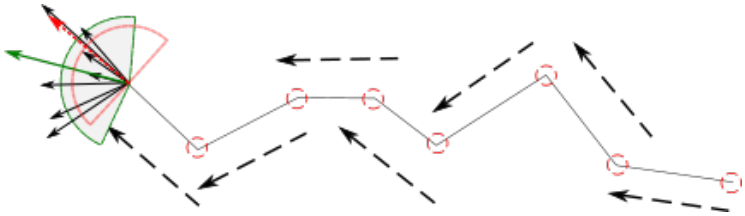


Figure 6.2. Memory constrained algorithm. The algorithm sums previous versors with a dumping factor α. The growth cone rotates and aligns itself with the angle weighted over the trajectory. The fluctuation due to normal distribution are attenuated and the correlation remains high despite the process has a large variance σ^2. In the figure the red semicircle and arrow represent the direction of last segment, the green circle is obtained with the memory algorithm instead.

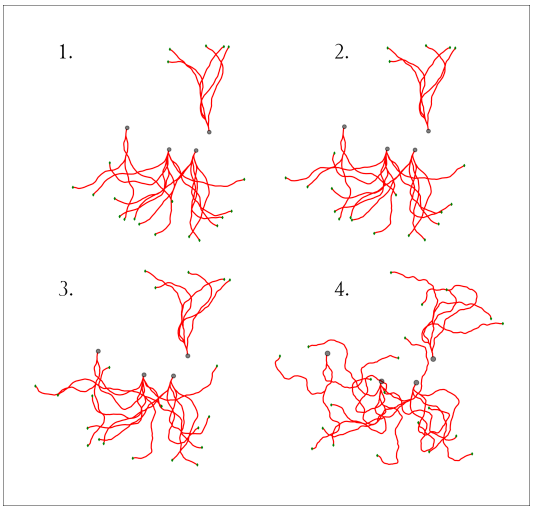


Figure 6.3. Neurons with different persistence length. In the upside plots a set of 4 neurons is growth with different parameters for the random walk, the branching model is Van Pelt and is discussed further in next sections. The persistence length decreases from 1 to 4. The adimensional parameters for Gaussian correlation and memory coefficient are set equal for simplicity. Plot assumes respectively values 0.7, 0.6, 0.5, 0.4.

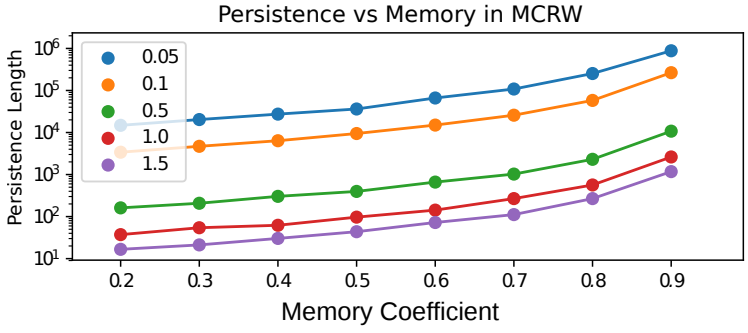


Figure 6.4. Persistence length is super exponential for MCRW The picture shows the dependence of the measured persistence length for the MCRW algorithm, the presence of memory makes the algorithm much harder to approach, even numerically. In log-scale for the ordinate, the persistence length appears super-exponential over α. Different colors represent different values of σ

For a scale inferior or comparable to the inverse of the Diffusion coefficient the GC presents ballistic pace, for larger scales the walk is diffusive. Assumed the elongation rate constant, v, a constant turning rate ν can be defined as a funtion of the time resolution Δt:

$$\nu = \frac{1}{v_{GC} \cdot \Delta t} \tag{6.12}$$

ν it is the number of rotation done by the GC in $1\mu m$. The diffusion process for the angle impose a linear equation for the Mean Square Displacemennt of the angle variable:

$$\langle \Delta\theta^2(l) \rangle = \sigma \nu l \tag{6.13}$$

From Eq. 6.6 and Eq. 6.5, looking for distance l much shorter than the persistence length:

$$C(l) = \exp(-l/l_p) \sim 1 - \frac{l}{l_p} \tag{6.14}$$

$$C(l) = 1 - \frac{1}{2}\langle \Delta\theta^2(l) \rangle = 1 - \frac{\sigma \nu l}{2} \tag{6.15}$$

$$l_p = \frac{2l}{\cdot \langle \Delta\theta^2(l) \rangle} = \frac{2 v_{GC} \cdot \Delta t}{\sigma} \tag{6.16}$$

$$\tag{6.17}$$

Eq. 6.15 is nothing else than the central limit theorem. By Eq. 6.16 the persistence length is analytically computable.
It is possible to verify its consistency measuring the persistence length with Eq. 6.4, which confirms the exponential decay. Fig. 6.5 resumes the results. The picture of the process in Eq. 6.11 is very clear, but the drawback of this method appears immediately when implemented: in order to have persistence length consistent with biological measured ones, the value of σ must be very little, i.e. less then $0.1 rad$. This becomes a relevant problem when upgrading to environment interaction: the GC interacts with the walls in a broader angle, around 180°, whether the walls interact as a potential on the angle distribution it is necessary that the probability of turning in the wall direction is more than one. This is to say in such model the GC will be blid to the walls except for a narrow angle.

Run and tumble To solve this issue we have followed the method used in NetMorph simulator: the turns are diluted in time with an exponential distribution. In the fashion of a run an tumble model.
The growth cone, has a certain probability to turn, depending only by the length of the step: $P_{turn}(l) = \frac{l}{\lambda}$. This model is called uniform turning rate (UTR) and produces straight intervals, with mean λ and exponentially distributed. In this case the coefficient $\nu = \frac{1}{\lambda}$ depends by the turning rate strictly. It is necessary to stress we are talking of rate in terms of spatial units.

$$P_{straight}(l) = \frac{1}{\lambda}\exp(-l/\lambda) \quad E[l] = \lambda \quad Var[l] = \lambda^2$$

$$l_p = \frac{2\lambda}{\sigma} \tag{6.18}$$

$$\tag{6.19}$$

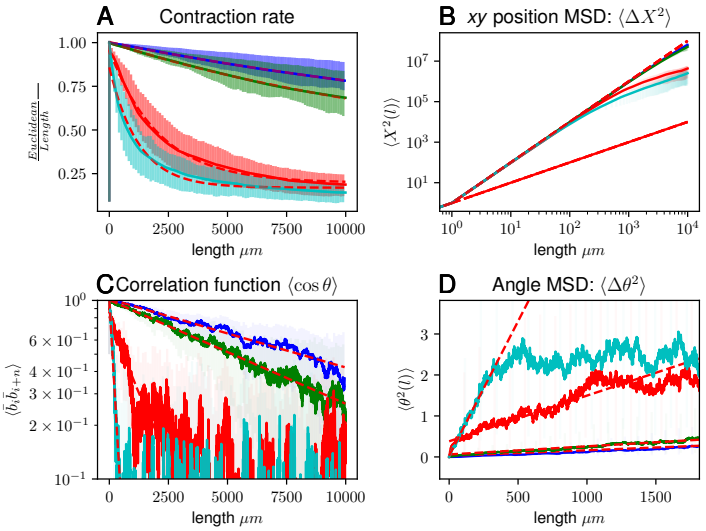


Figure 6.5. Correlation and contraction measures for different models A set of similar statistical measures is applied to navigation models presented. The colors stand for different models:
Light blue is simple diffusive RW, Red is the self referential force model proposed in [Memelli et al., 2013], green is the Run and Tumble and dark blue is the MCRW devised by us.
A The contraction rate is the ratio between the distance from the soma and the length of the neurite, it converges in diffusive regime as discussed in the AppendixA. **B** The *xy* MSD says the caverage distance from the soma as a function of length, when the regime moves from balistic to diffusive it changes from quadratic to linear. **C** Shows the correlation between segments direction, it is a numerical implementation of Eq. 6.4, it goes to zero when diffusive regime starts. The persistence length can be obtained by a tfit of this measure. Eventually **D** present the angle variable MSD, it diverges very slowly in UTR and MCRW, while surpasses π in few steps in other models. The angle variable diffuses in all the models and the slope should reveal the persistence length also, the slope is related to the D_{coeff}. Detailed explanation of methods in the Appendix A.

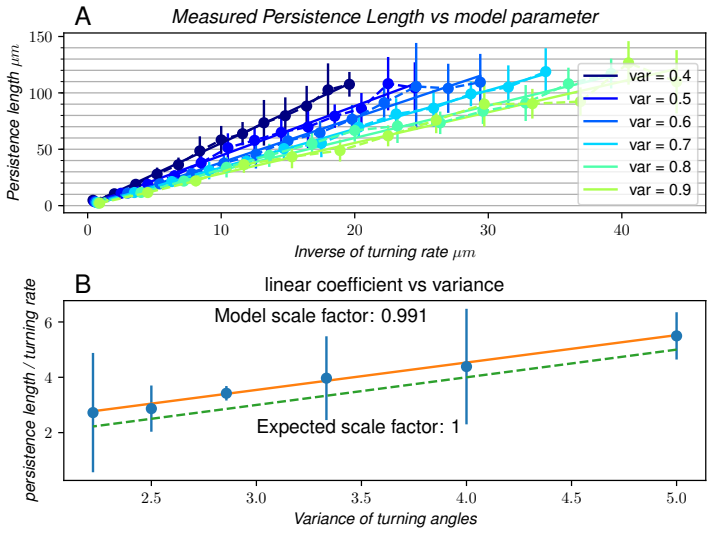


Figure 6.6. Persistence length as a function of turning rate In this two pictures we verify the linear equation for the Uniform Turning Rate model 6.18. Focusing on the concept of turning rate shifts the attention from the speed and the time resolution, to the actual spatial realization of the process: measuring the turning rate in micrometres instead of second is a robust definition that is invariant in respect of the resolution and the time-step; even it is well defined for neuron images. **A** shows the persistence length measured for varying parameters σ and λ. The persistence length is measured as the ensemble average of $\langle cos(\theta) \rangle$. The data are produced by the NetGrowth simulator itself. In **B** the angular coefficient obtained above is plotted over the expected value with an excellent precision. The plot confirms that for the UTR algorithm the effective persistence length can be arbitrarily set by θ and λ.

With the last equation the model was largely enhanced: the persistence length is now independent by other quantities, like growth cone speed or resolution time, and can be fixed, case by case, by the parameter λ. In Fig. 6.6 the linear dependence of the persistence length is shown for variation of both σ and λ.

6.1.4 Methods an Validation

To study the algorithm a vast amount of experiments with single growth cones was run, they make up a statistical ensemble. Hence the percentile distributions were computed and the fits too. To study real neurons we have looked at NeuroMorphoNeuroMorpho database database, we have used rat retinal neurons since any other two dimensional neuron was available and there weren't SWC files of cultured samples. To measure their properties we have decomposed each branch tree into single paths and computed ensemble average.
This approach has some inner limits: study stationary properties of non equilibrium process is hard, even more if the dynamics is lost and a 2D static image is the only track of the process. Furthermore the axon is just one and was impossible to study its statistical properties on the reduced set we disposed. With the dendrites we were luckier and a complete characterization was performed.
Unfortunately the neuron growth in cultures are not on the database, which concen-

trates on *invivo* samples: to study 2D neurons we had to adoperate neuron from the retina of rats. These neurons are slightly different from those present in neurons cortical area, used for the cultures. Anyway a sophisticated tool to reconstruct the neurite from the SWC file has been done and can be used in future with effective data. The principal measure remains the persistence length that can be measured from mature neurons or even in laboratory with microtubule dissociation as evinced in Fig. 6.1. Some of the analysed neurons are reported in Fig. 6.7

6.1.5 Conclusion

The analysis of the navigation mechanism for the growth cones, and the simulation of it, has required a deep insight into bi-dimensional stochastic processes. The properties required for the simulator have been outlined in the first section and the algorithms have been evaluated over them. Eventually a modified diffusion process has resulted enough simple to be analytically solved still meaning full for the scopes. The benchmark of the model are cultured neurons, but relevant data were not found on the Web and so a precise verification is postponed. In Fig. 6.9 some neurons built with UTR model varying the rate parameter are plotted. Further model for environment sensing and direction election will be studied in future, a particular attention will be reserved to self-avoiding walks.

6.2 Growth in a heterogeneous environment: sensing the surroundings

The way the growth cone sense the environment biologically is complex and very hard to reproduce in a simulation, it's roughly described in previous section and more accurately in [Roth et al., 2014].
We have summarized over the whole biological process (focal adhesion, pulling, etc...) and considered the environment interaction as a bimodal process: the obstacles can attract or repel the GCs. In case of attraction the probability the active unity will move towards the obstacle is higher in respect to a growth cone indifferent to the environment, opposite otherway. The growth cone is kept into the cultures walls and cannot overpass culture's shape, the obstacles are walls in the culture that occlude the access to growth cones, as black cells in a crossword puzzle. The sensing operation is done at each time step, it has to be unexpensive from a computational point of view and deactivable when it isn't required.
The model we proposed was first sketched in R.Renaud [Renault, 2015], we have changed it to fit with the computational requirements of NetGrowth and to integrate with the random walk models previously discussed.
The wall-filopodia interaction is powered by a geometrical engine. The environment model presented in this section allows to simulate complex culture and behaviours as the one described in Fig. 6.10.

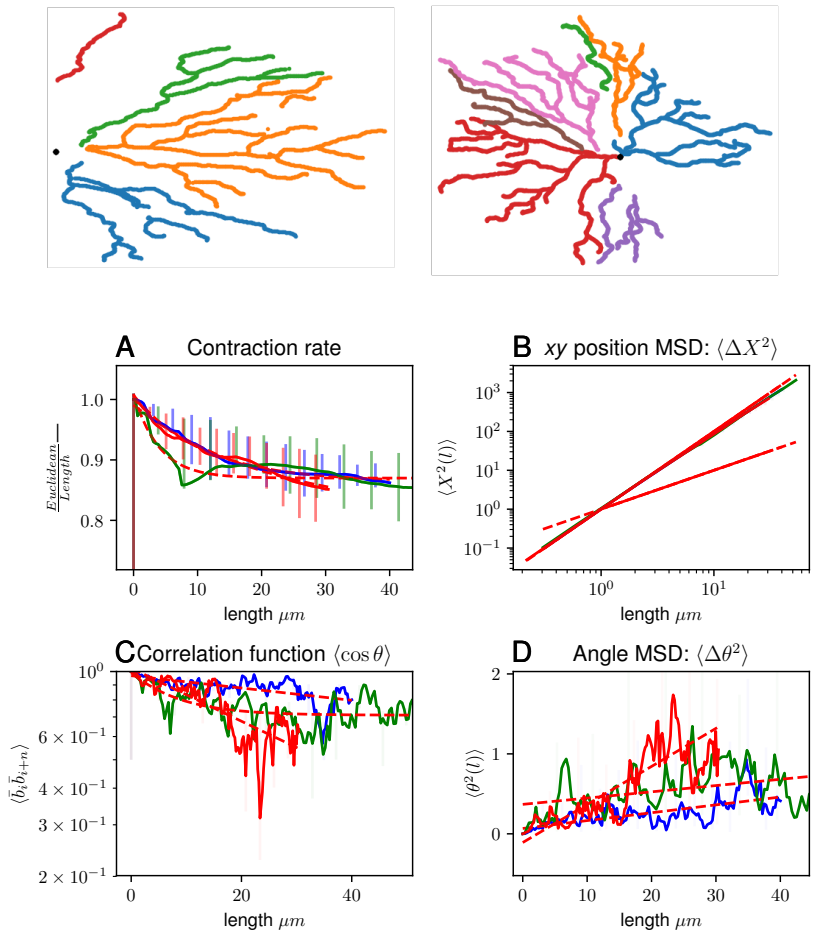


Figure 6.7. Characterization of dissociated neurons A set of neurons from NeuroMorpho database are measured to extract the characteristic persistence length. The images, in SWC format (see Appendix F.2) have been processed and reconstructed to analyse the property of the whole neurite. Ech neurite has been decomposed in shorter and longer branches and the last one have been averaged for the salient measures. The neurons plotted are two of the analysed ones. Unfortunately NeuroMorpho has not 2D reconstructions, hence we had to use retina neurons and not cortical cells or interneurons. In the lower set of pictures we reproduce the standard measure to calculate the persistence length, this is attested around 50 micrometers. We cannot affirm this measures are sound, indeed the persistence length appears to be much longer and the diffusive regime far. Hence we cannot say whether or not Random Walk regime will effectively appear: In picture **C** the correlation results strong after 40 μm and we cannot predicts it will go to zero or not.

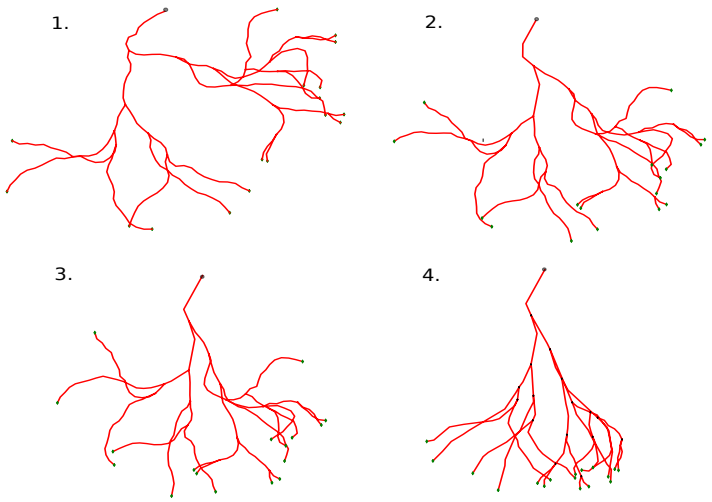


Figure 6.8. Varying parameters for the UTR model In the pictures from 1 to 4 a neuron is generated with an increasing persistence length.
The simulator can reproduce a vast amount of neuronal structure, varying the branching rate, the speed and the persistence length. The user who wants to simulate a precise neuron has to fit his experiment or find data of the type of neuron he wants to simulate.

6.2.1 Interaction with environment as probability functional convolution

The model implemented in NetGrowth to sense the environment recalls to growth cone biology: it checks the superposition between a filopodia and the wall. When the walls are distant enough that the GC can step towards it within a time step the *wall affinity* is considered and the probability of going that way is modified, increased or decreased. In case the wall is to close to the growth cone central domain the probability is set to zero and the GC is forced to go in another direction, see Fig. 6.11.
The first characteristic this model presents is the discretization of space, the direction the growth cone can move to depends on the angular sensibility of filopodias. The probability distribution becomes a histogram and environment perception is coarsen. To simplify this issue the filopodia density is taken uniform and maintained in a certain angle ahead the cone.
The probability is convolved with the probability distribution computed in the frame of random walk, in such a way the stiffness and other properties of growth cone spatial dynamics are kept by the convolving probability distribution, while the environment can pull the axons by its side and affect the overall picture. This model has some drawbacks which are introduced and solved in next sections.

6.2.2 Implementation in NetGrowth

In order to implement the model in NetGrowth we needed to have a simple picture of the sensing unit and to fit it inside the simulator growth cone unit. The easiest

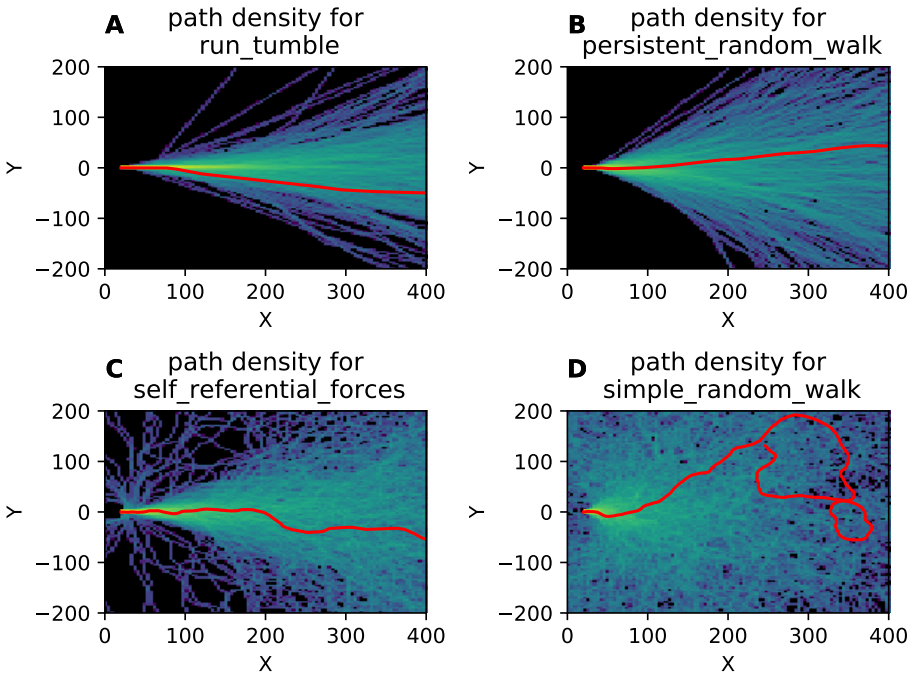


Figure 6.9. Density distribution for different models of GC navigation To choose an appropriate model for the growth cone navigation we have evaluated the outcome of some *classical* algorithms and created a new one. In pictures we show the average path, and a sample path in red, of the outgrowing neurite. Pictures **A** and **B** appear similar, yet the former (UTR or run tumble) is resolution invariant and the latter (MCRW or persistent random walk) is not. The MCRW is governed by three parameters, that makes very hard to predict its behaviour for a changing time resolution. Picture **C** reproduces the algorithm proposed by [Memelli et al., 2013], although it is demonstrated it produces relevant morphologies it has four parameters which are not directly related to a possible experimental measure, also the algorithm was very hard to tackle analytically and to predict its time-resolution dependence, furthermore it is not a generalized Random Walk and it is not sound to speak about persistence length in this case. Eventually in **D** a simple random walk is illustrated. This process does not produce plausible morphologies: it is regulated only by the diffusive coefficient, if it is very low the growth cone goes straight and ignore the surroundings, other way it turns and self-crosses repeatedly.

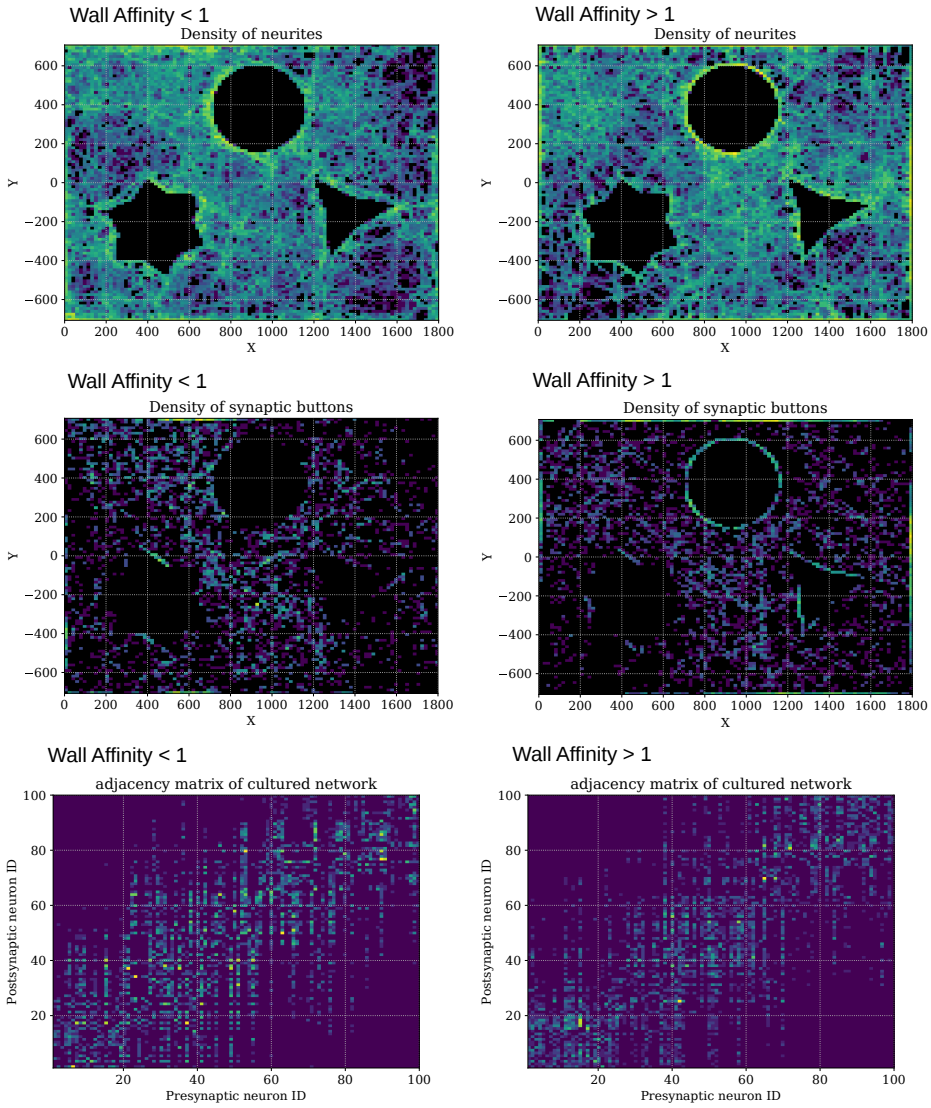


Figure 6.10. Complex shapes and environment interaction. The picture portraits a culture with 100 neurons simulated with NetGrowth; neurons with negative wall affinity are presented in the left column, while positive wall affinity neurons are portrayed on the right. The *repelled* neurons avoid the walls and spend most of their length in away from the borders, wit is the opposite for *attracted* neurons, this can be noticed in the top pcitures that showes the density of dendrites in the culture.The picture shows the variability that the environment interaction model can produce with a single parameter, the *edge affinity*. The second row present the location of synaptic buttons generated with the *intersection* method. The most of synapses of *attractive* neurons appear along the edges, while are sparse in the culture in the *repelled* case. This configuration has an evident effect on the adjacency matrix: the left graph appear more connected and the neurons have synapses with neurons farer then in the right case, this consequence should be verified experimentally and could be an experimental evidence for the algorithm validation.

The wall-interaction algorithm is among the completely new features introduced with NetGrowth simulator. It is possible to affect the Growth Cone navigation - changing both the speed, the persistence and the elongation threshold - as a function of the substrate it stands on. This feature becomes particularly relevant in respect of the several experiments conducted on mixed substrates.

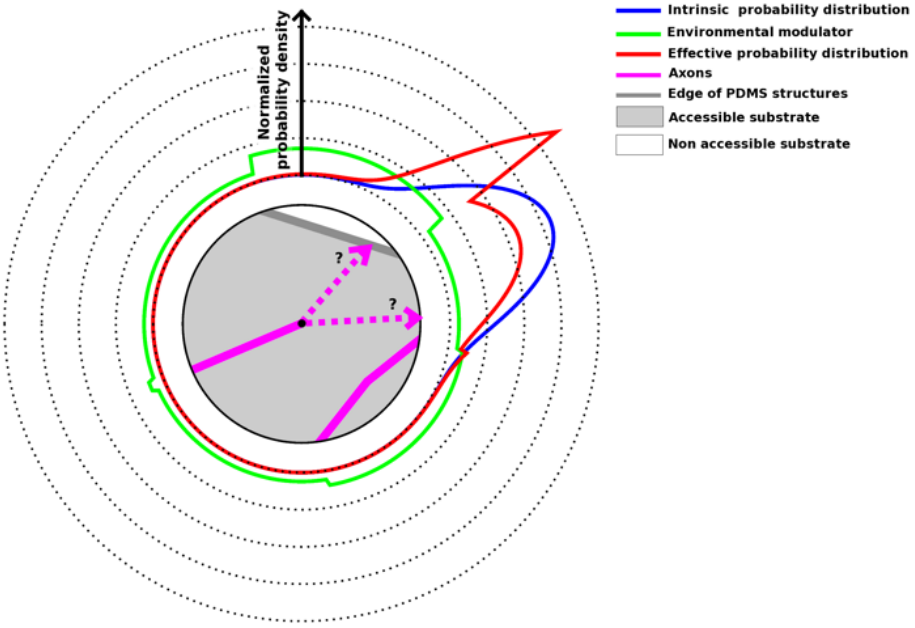


Figure 6.11. Interaction as functional convolution of probability The growth cone senses the environment with its filopodia (whose length is tunable) and changes the distribution of the navigation model: The probability of going in a certain direction, which is computed within the RW or RT model, is convolved with the potential offered by the environment, be it attractive or repulsive. In the picture, obtained by [Renault, 2015] the intrinsic probability is represented in blue, while the environment potential is pink. The result convolution is drawn in red. It is a continuous distribution which is reproduced in the simulator. On this purpose we have tested the model varying the number of discrete angles. The wall on the upper side acts an attractive effect, stronger than the fasciculation with the other axon on the right.

way is to use a geometry engine and let it compute the intersection between wall and filopodias, then create the distribution histogram, in each filopodia direction, and choose stochastically one of them.
The sensing happens if the distance between the GC and the obstacles is less than the filopodia length, which is estimated 20 microns [Cheng et al., 2002], the potential of attraction with the wall can change depending on the distance, i.e. be stronger if the wall is distant more than half of filopodia and even be repulsive in the proximity of the cone central domain. The parameter to define the number of filopodia in the GC is *angular resolution*, their length is identified with *filopodia length* and the attraction is regulated by the *edge affinity*.
The model interacts with the random walk model: the direction is chosen after both the distribution are computed. The limitation introduced is to use a discrete set of angles for the direction model instead of a continuous variable, we have performed some tests to verify this drawback is irrelevant, given the angular discretization is enough dense.

Renaud model for probability election, from continuos to discrete probability

Let's introduce the model itself with a step by step description than can be traced, line by line, with the code in 6.1. The first action of the GC is to sense the environment computing the intersection by its filopodias and the wall. The filopodia are numbered and this number is depends by the time resolution. In function *compute pull and accessibility* the probability for each filopodia-direction is initialized to one and then increased or decreased by the *edge affinity* parameter in case of contact, maintained otherwise. This step produces an histogram, each direction has a multiplicative factor that can be grater (affine) or lower (repulsive) than one.
In a second step the histogram is convolved with the Gaussian distribution of the angle, the procedure is detailed in Fig. 6.12 At the end we have a histogram of values, doesn't matter if not normalized, and we have to choose one direction among them, with a weighted probability.

Listing 6.1. Implemented algorithm for environment sensing

```
void GrowthCone::compute_pull_and_accessibility(
std::vector<double> &directions_weights, mtPtr rnd_engine)
{
//set all filopodia to 1
for (int i = 0; i < filopodia_.size; i++)
{
directions_weights.push_back(1.);
}
while (all_nan and not stuck_)
{
   // initialize directions_weights
   // test the possibility of the step (set to NaN if position is not
       accessible).
   //the filopodia are checked within a [-4 sigma, 4 sigma] interval,
       sigma is the only physical parameter here.
   kernel().space_manager.sense(directions_weights, filopodia_,
```

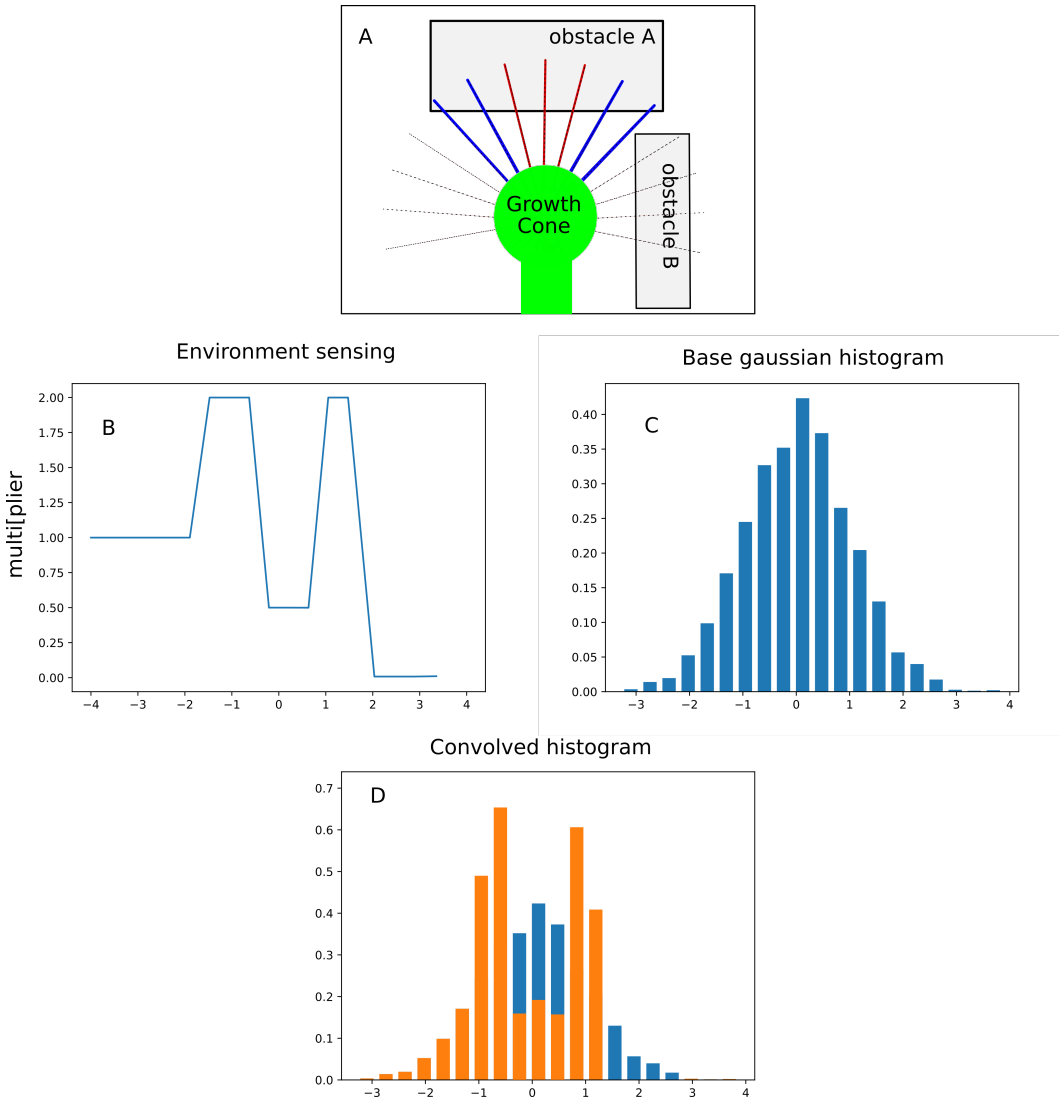


Figure 6.12. Environment aware directions' distribution The picture (**A**) offer a schematic but consistent idea of the NetGrowth model for environment interactions, the GC in green elongates its filopodia towards the exterior, they *touch* the surroundings and return the information to the GC central domain. Depending on the parameters chosen the filopodia can be attracted or repelled by the obstacle. In this case the three central filopodia (red) are to close to the obstacle A and they are repelled, laterals (blue) are attracted instead, while the four filopodia on the right are to close to the obstacle B so as to set the probability of going that direction to zero. The environment interaction is summarized in picture (**B**) which multiplied by the normal histogram (**C**) produces the convolved histogram (**D**). The last histogram is the baseline probability to compute the next step, the direction is drawn from the distribution with a reservoir sampling method as discussed in its section.

```
   geometry_.position, move_, sigma, move_.module, 1., nan(""));
   //if any direction isn't available set 'all_nan' variable
   all_nan = allnan(directions_weights);
   //this is the advantage of variable sigma method: whether the cone
       looks stucked try to enlarge the sigma and accept broader angles.
   if (all_nan)
   {
      if (abs(move_.sigma_angle - M_PI) < 1e-6)
      {
         // completely stuck
         stuck_ = true;
      }
      else
      {
         // increase sensing angle and reset weights
         move_.sigma_angle = std::min(2 * move_.sigma_angle, M_PI);
         for (int i = 0; i < directions_weights.size(); i++)
         {
            directions_weights[i] = 1.;
         }
      }
   }
   // if the cone isn't stucked feel the sourroundings and fill the
       discrete distribution.
   else
   {
   // test the presence of walls to which the growth cone could
   // be attracted
      kernel().space_manager.sense(directions_weights,
         filopodia_, geometry_.position, move_,
         filopodia_.finger_length, 1., filopodia_.wall_affinity);
   // check stronger interaction if filopodia is wall is close enough
      if (0.5 * filopodia_.finger_length > move_.module)
      {
      kernel().space_manager.sense(
         directions_weights, filopodia_, geometry_.position,
         move_, filopodia_.finger_length * 0.5, 1.,
         filopodia_.wall_affinity * 2);
      }
   }
}
}
void GrowthCone::compute_intrinsic_direction(
std::vector<double> &directions_weights)
{
for (int n = 0; n < filopodia_.size; n++)
{
directions_weights[n] *= filopodia_.normal_weights[n];
}
}
```

Sampling discrete distribution

Once the directions' histogram all we need to do is to choice a direction. To draw an element from a weighted list there is a naif and robust method: first we sum up the weight of each element and obtain the normalization factor, then we draw a number from uniform distribution, multiply it for the norm and, ordering the elements somehow extract the one whose interval contains the drawn number; in a more figurative way we will assign each element an area proportional to its weight and choose a random point on the plane. With this simple method the normalization isn't required and this is useful to simplify the method of variable sigma that we are going to describe.

Geometry with GEOS and variable sigma method

To compute the intersection we have used a simple and powerful geometry library for C++. GEOS[2] allows for python binding, is thread safe in parallelellised codes and very simple and efficient in respect to more famous projects, which often require a huge modification to the code to include them into. The code we have used is very easy: we create a bunch of straight lines, as many as the number of set filopodia, where one of the two extremal is the centre of the filopodia and the other is the projection of the angle at the sensing distance. Once the filopodia are created the engine evaluates whether they intersect or not the environment, the latter is a geometrical object passed to GEOS at the simulation kernel initialization. Further details will be offered in the documentation of NetGrowth. The interesting point in such a paradigm is the method we used to test at the proper angle and convolve the environment distribution with the intrinsic Gaussian without compute the Gaussian at each step, see Fig. 6.12.
We have introduce a simple and smart method we have called 'variable sigma'. Indeed we use this fact: a set of points where the normal distribution can be evaluated $\mathcal{X} = [x_0, ..x_n]$ are self similar to the distribution computed over the same set of point multiplied for a fixed constant: $\mathcal{X}' = [\sigma x_0, ..\sigma x_n]$ that is

$$\exp(X) = \frac{\exp(X')}{\sqrt(\sigma)}$$

Such a property of the normal distribution makes very easy and simple the computation, indeed the whole operation can be computed over a distribution with $\sigma = 1$ and the normal weights don't require to be computed again. The useful of the method comes up when the GC is stucking in a corner too, after few runs with environment distribution set to zero for each filopodia the sigma augments and let the GC to get out of the corner.

6.2.3 Turning angles

A first check of the quality of the algorithm can be devised reproducing *in silico* the experiment discussed in Fig. 4.8. When thew wall affinity is positive the GC will adhere the wall, it can follow the edge when it is straight and this will be either in

[2]Geometry Engine Open Source, https://trac.osgeo.org/geos

agreement with the intrinsic to avoid turning, due microtubules' stiffness. Hence when the wall turns abruptly if it will follow the edge or keep going straight will depend by the strength of attraction and its counterpart, the microtubule stiffness, which we have parametrized with the *persistence length*. In Fig. 6.13 we discuss an experiment with NetGrowth with fixed *persistence length* and variable *wall affinity*. It reveals there is a strong dependence by the latter factor. Devising an *in vitro* experiment ad hoc will be possible to relate the parameter to some real properties of the neurons.

6.2.4 Mechanical diode

The second benchmark for the model is offered by the mechanical diode devised by R.Renaud. Such a device can be very useful in a culture because it selects the direction of axons (and the connectivity then) and doesn't require any chemical treatment for the axon guidance, that is, can be used massively for an unsupervised culture, the experiment is discussed in detail si in Fig. 4.10. The diode is tested in Fig. 6.14 and offers quite good results. The study is purely qualitative since the dynamics of the axons i much more complex than the one reported. Further experiment, indeed, reveal an high correlation between the narrowness of the walls and the speed of the GC. Such evidence lead to a complicate model, where the environment interaction module is interacting with the elongation module. Further studies are required in this direction and for the moment the qualitative overview is enough.

6.2.5 Conclusion

The module for environment interaction has been qualitatevely tested with experimental data, a correct test would require a dedicated experimental setup. However changing the parameters produce relevant and comprehensive behaviours. The diode implemnted at IPGG was tested and offered consistent results. On the other hand experiments at IPGG, and other authors too [Bak and Fraser, 2003], describe the importance of axon fasciculation both in two and three dimensional cultures. Fasciculation isn't yet included in the simulator since a more sophisticated computation is needed for. The group planes to work on a new model to include this feature.

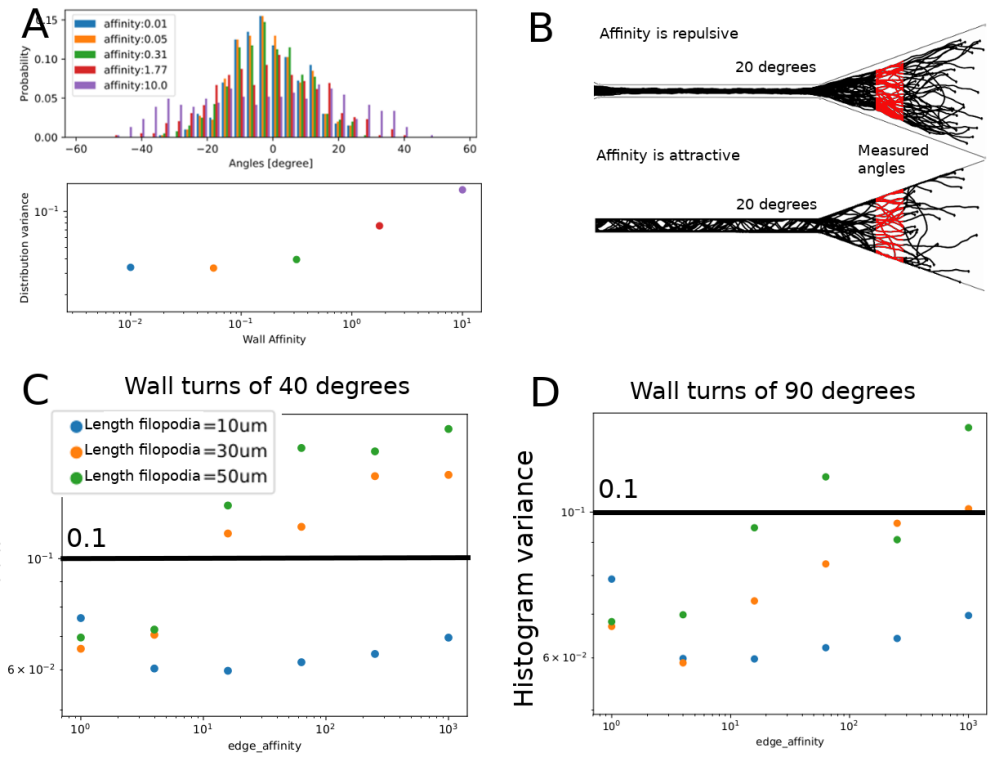


Figure 6.13. Growth Cones' response to turning angles In this picture we present some quantitative data that can be matched with *in vitro* experiments. The neurons are growth with the simulator in the confinement illustrated in picture **B**. Different set of parameters have been tested, varying both the length of the filopodia, i.e. the distance up to the wall is sensed, and the affinity with the wall itself. For each parameter the average angle of the outgrowing neurite was measured, the angle is averaged on the spatial area illustrated by the red stripes in **B**. It is shown in the histogram in box **A** that the affinity affects sensibly the angle distribution, the histogram represents the amount of neurites with a certain angle, it is indeed the distribution of the angles. The mean value is expected to be zero for the symmetry of the confinement, hence the variance was calculated and the values are plotted as function of wall affinity in the bottom plot of **B**. It is evident that for an increasing wall affinity the distribution flatten, and even the tails are more populated then the center. Pictures **C** and **D** show the effect of different filopodia length: it changes the distribution variance too. When the filopodia is short, i.e. 10 μm, the wall is not sensed by the growth cone and it is indifferent to the wall affinity parameter. In the two picture two different angles of turning are tested. The black line shows the most of growth cones does not follow a sharp turns of 90°, neither for high affinity and longer filopodia.

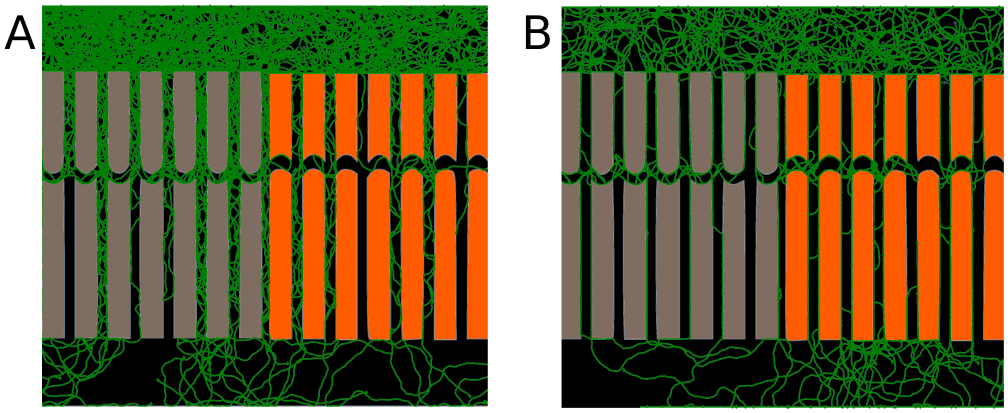


Figure 6.14. Mehcanical diode tested with NetGrowth This figure qualitatively verifies the diode implemented at IPGG, reproducing the result with NetGrowth. In the picture the orange elements on the right are diodes up-to-bottom while the grey elements are bottom-to-up.
In picture **A** the effect of a non interacting model (*edge affinity* is 1.): the neurons are kept inside the culture but any attraction nor repulsion from the walls isn't present. We see the presence of many axons in the bottom of the picture, decorrelated with the presence of the diodes. In picture **B** the GC-wall attraction is switched on (*edge affinity* is 20.): the correlation with the diode is very high, most of the axons in the lower chamber come from the right (up-to-bottom) diode, as expected.
The other parameters are identical for both the pictures: the persistence length coefficient is set to 20. while the filopodia are long 10 μm, σ is 8.2 degrees and the angular resolution is 40.

Chapter 7

From single Growth Cone to Neurites

Contents

Neurons connect each other through growth cones, and these connections have to be multiple to turn on the complex behaviour of brain tissue. We have outlined the strategies for this objective are many, but branching is the most widespread for all the neurons. On the other hand, from the descriptive point of view branching is the *de facto* nature of neurites: a biologically relevant simulator must reproduce this mechanism as close to the real phenomena as statistic measures can evaluate. The branching itself leads to more complex processes that affects the whole neurite: competition, fasciculation, retraction and pruning etc... The wide variety of these events and mechanism prevents from an accurate description of the arborization, but it's possible, as generally is done in physics, to stress the relevance of some aspects over the others. In a more abstract way we would say that the best model is the simplest one that is enough complex to describe the actual phenomena, all further complications are avoidable and can cloud the interesting behaviours. An example of such choices is the following: in the first stage of Netgrowth realization we decided to describe the Actin Waves, an extraordinary phenomenon that shapes the neurite initiation phase, after a while we understood the minor relevance of this

in branching formation, and preferred to remove from the simulation. We chose to implement those mechanism that can reproduce a complex arborization as the one showed in Fig. 8.4.

First the Growth Cone is enabled to split, the split is forced to respect a volume conservation law. The neurite can also branch laterally and originates a new GC. Eventually GCs compete each other and while some elongate some other retract. When retraction is continuous the branch is eventually pruned.
This chapter is divided in three sections: the branching mechanism, in terms of events' probability is described in the first section, here we present an adaptation of Van Pelt model for arborization, and a uniform branching model for lateral branching. The second section introduces the diameter to compute branching angles and branches' diameters. Once the neurite with multiple branches is introduced it is important to predicts their elongation rate, the speed of growth cones; this aspect is tackled in the third section, here we introduce a brand new model for critical resource based competition, it is defined and studied analytically. Eventually the produced neurons are evaluated with standard measures for neuronal morphologies and confronted with other academic results.

7.1 Modelling branching processes

Here the NetGrowth algorithm for branching events are accurately described, first the Growth Cone splitting, and later the shaft branching. I have not produced an original model in this case, rather I adapted a successfully tested one to implement it efficiently in NetGrowth, maintaining the relevant physics behind.
Let's introduce briefly one of the simulation constraints that influenced the implementation of both these models: A general request for the simulator is to reduce the operations wherever is possible, even more to produce as few stochastic variables as possible; this constraint is particularly relevant in case of branching with BEST model. When the occurrence of branching is governed by a rate at least two approach are possible. A naïve method suggests to compute the branching probability and check whether it happens, for each step; that is to generate a random variable and test it is less then a certain threshold. In such a scenario the number of operation grows with the number of growth cones and, even worse, is influenced by the resolution time. A valid approach, instead, consists into switching from the branching rate to the interval distribution, pre-computing the time next event should happen. Hereafter to predict the branching event and apply it only at the forecasted time-step saves a huge quantity of computers' clocks.
In the following studies the rate depends on the state of the neurite only, i.e. the number of growth cones, the age of the neuron, and so on, hence the interval distribution can be precomputed for each neurite given the state will not change during the interval itself, e.g. recompute it when a new growth cone rises for other model or disappears for pruning. With this approach we can reproduce exactly the BEST model [van Pelt et al., 2003a] and whichever branching mechanism whose interval distribution is known. For the interstitial branching a similar approach is pursued.

7.1.1 Growth cone splitting, reversing Van Pelt distribution

To estimate an interval distribution consistent with BEST model[1] introduced in Eq. 7.1, we have started from the branching rate:

$$p_i(t|n(t)) = \mathcal{N} n^{-E} 2^{-\gamma_i S} D(t) \tag{7.1}$$

with the normalization factor (7.2)

$$\mathcal{N}^{-\infty} = \sum^{n(t)} 2^{\gamma_i S} \tag{7.3}$$

where $p_i(t|n(t)$ is the rate of ith Growth Cone splitting (namely the transition rate from a state with n to $n+1$); the function $D(t)$ is a baseline branching distribution, exponential decreasing, determined by experimental data: this distribution states for the diminished probability of observing a branch event in mature cultures, from [van Pelt et al., 2003a] we set $D(t) = A\exp(-t/\tau)$. The other parameters are E which define the competition among growth cones and S which accounts for tree asymmetry [2], the last one is less important for this description because the normalization factor ensure the branching rate independent by the parameter S. The average number of tips, i.e. GCs, in the neurite is governed by the differential equation:

$$\frac{dn}{dt} = n(t) p_i(t|n(t)) \tag{7.5}$$

We can integrate Eq. 7.1 over all the possible segments and obtain the total rate of branching $p(t|n(t)$ as:

$$p(t|n(t)) = \sum_i^{n(t)} p_i(t|n(t)), \gamma_i) = D(t) n(t)^{-E} \tag{7.6}$$

$$\frac{d\langle n\rangle}{dt} = \langle n(t)\rangle p(t|n(t)) = D(t)\langle n\rangle^{1-E}(t) \tag{7.7}$$

And set the boundary condition $n(0) = 1$ which corresponds to start with one Growth Cone. The Eq 7.7 can be solved as a separable first order ODE and we can explicit the $\langle n(t)\rangle$ as a function of time and branching parameters A, τ, E

$$\langle n\rangle(t) = (\frac{AE(1-e^{-t/\tau})}{c} + 1)^{1/E} \tag{7.8}$$

[1] One of most cited branching model, it is a fully phenomenological model whose parameters can be obtained from statistical measure of mature neurons

[2] For a tree α of degree n:

$$A_p\left(\alpha^n\right) = \frac{1}{n-1}\sum_{j=1}^{n-1} A_p(r_j, s_j) \tag{7.4}$$

is the mean value of partition asymmetries in the tree, which, at each of the n bifurcation points, indicate the relative difference in the number of terminal segments r_j and s_j in the two sub-trees emerging from the jth bifurcation point. The partition asymmetry A_p at a bifurcation is defined as:

$$A_p(r,s) = \frac{|r-s|}{r+s-2}$$

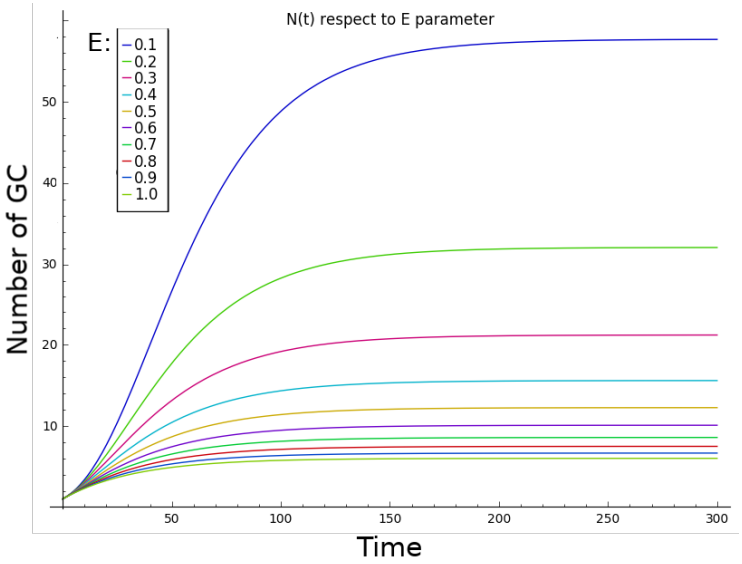


Figure 7.1. Expected number of branching events Varying the competition parameter the number of growth cone in the final tree changes, this dependence agrees with the competition model presented lateron. The number of branching events is regulated by the differential equation in Eq. 7.8. The picture presents the number of total branches expected.

$\langle n(t) \rangle$ is a continuous, real, variable accounting for the average number of growth cones. Its evolution depends by for varying the competition parameter is presented in Fig. 7.1.

A formal approach we should compute the probability of having n events, and then compute the average time between two events, given the parameters $\langle \Delta t \rangle (A, c, E)$ and its distribution. This process it is hard to study correctly, indeed the probability changes both in time and for the discrete component $n(t)$ thereafter a Fokker-Planck description would require a left-side repulsive barrier evolving in time. Anyway, the interesting distribution is not the $n(t)$ at the end of the growth but the events interval. A simpler but less rigorous approach is to estimate the amount of time required to increase the number of GCs, given $p(t|n(t))$, the branching probabilities of the individual terminal segments per unit of time. Let's proceed by step and assume $D(t)$ is constant in time, that is the rate of branching is uniform during the growth, this rate, ν, depends only by the number of growth cones and is constant in the interval between two branching events. In the limit of a short interval Δt[3], defined $P(X(t) = n) = P_n(t)$ with X the number of branching events, the following equations hold:

$$P_n(t + \Delta t) = P_0(t)(1 - \nu \Delta t) \tag{7.9}$$

$$\dot{P}_0(t) = -\nu P_0(t)$$

$$P_0(t) = P_0 e^{-\nu t} \tag{7.10}$$

This distribution is the widely used in physics, and neuroscience, and it is an exponential decaying probability distribution with mean value $\frac{1}{\nu}$. Up to here the

[3] $\Delta t \cdot \nu \ll 1$

interval distribution can be computed easily and is exact. Let's see what happens in our case study which is explicitly time dependent. The correct equation to solve, for the first event, becomes:

$$\dot{P}_1(t) = -\nu(t)P_1(t) \tag{7.11}$$
$$= -Ae^{-ct}n^{-E}P_1(t) \tag{7.12}$$
$$P_1(t) = P_1(0) \cdot e^{-\int_0^t \nu(t)dt}e^{-1} \tag{7.13}$$
$$\tag{7.14}$$

which becomes, for $P_1(0) = 1$:

$$P_1(t) = e^{\frac{A}{c}(1-e^{-tc})} \tag{7.15}$$

The first relevant aspect of the Eq. 7.15 is the probability of having only one branch does not disappear for large time. This fact can intuitively explained in this way: whether the baseline has a short characteristic time, the rate decays so fast the any branching event can happen. Therefore the interval mean time is:

$$\langle \Delta t \rangle = \int_0^\infty t\mathcal{N}e^{\frac{A}{c}(1-e^{-tc})} \tag{7.16}$$

For second and high orders interval the computation becomes tricky, indeed the baseline is an independent process. The explicit time dependence prevents to compute interval distribution with simple methods, a correct computation should comprehend the interval distribution of all previous extracted intervals. In principle, after $p_n(t)$ is obtained the *first return theory* [e Parisi, 2010] can be used, hence first return probability $f(t)$ computed and eventually the average interval and its distribution. In order to make the computation feasible, and avoid a demanding analysis, the time dependence can coarse grained and substituted with average values:
The methods can be outlined this way: First, the rate defined in Eq. 7.6, is approximated by its value at the beginning of the interval,

$$D(t) \to D(t_0) = \exp(-t_0/\tau)$$

, this approximation is valid if the mean interval obtained this way is much shorter then the characteristic time. Second the problem is not solved in general but dynamically: we are, in principle, interested to the interval distribution of the average process, but, to forecast the next event with the proper distribution is enough for the simulator. Hence the number of growth cones is assumed as a parameter of the process and the interval distribution, given n events happened, becomes:

$$\nu(t_n, n) = D(t = t_n)n^{-E} \tag{7.17}$$
$$\langle \Delta t(n) \rangle = \int_0^\infty t \exp(t\nu(t_n, n))dt$$
$$= A^{-1}e^{t_n/\tau}n^E \tag{7.18}$$

where t_n states the time of the nth event. The constraint introduced above is respected for:

$$\tau > \langle \Delta t_(n) \rangle \tag{7.19}$$

$$\ln(\tau) > \frac{t_n}{\tau} + E \ln(n) + \ln(A) \sim \frac{t_n}{\tau} \tag{7.20}$$

$$\tag{7.21}$$

This distribution was implemented in NetGrowth, the results are confronted with the BEST model in Fig. 7.2 After the interval time is extracted the exact branch where the split happens is selected by a weighted choice basic algorithm. The weight depends dynamically by the branch order in the tree and by the symmetry parameter S introduced above.

7.1.2 Lateral (interstitial) branching

For the lateral branching a similar approach is proposed following the results in [Szebenyi et al., 1998]. Each segment has a certain probability to generate a new growth cone, which depends only by a local excess of actin. These excess can be due to fluctuations or growth cone pausing. The new branch appears laterally and starts a new growth cone. The GC so generated is equivalent as it is appeared after a GC split event. For a correct characterization and simulation of the process we investigate the explicit time dependence fo lateral branching events, and the spatial distribution of new GCs over the branch length. The cited paper describes the process accurately and reports numerical results in terms of branching rate and distance from the leading growth cone, some relevant data are represented in Fig. 7.4. Following the success reported in previous section the same method is applied: the rate is assumed to decay with a certain function

$$D(t) \sim t^{-\alpha}$$

, which can be fitted by data. Hence the expected interval Δ_t is computed assuming the rate constant in and the branching event time is drawn by an exponential distribution (fixed rate assumption). With this method the expected number of branching events, in a certain interval, is coarsely governed by the value of $D(t)$ in the interval itself. As above this method presents severe flaws when the function $D(t)$ changes rapidly in respect to the mean duration of the interval Δ_t. But this is not the case for the data reported in [Szebenyi et al., 1998]. The spot where the branch rises is chosen by an exponential decaying distribution, as explained in Fig. 7.4. An example of dendritic tree generated by this method is presented in Fig.7.3.

7.1.3 Conclusion

The branching models for Growth Cone splitting and lateral branching have been adapted to minimize the computational costs. The exact estimation of the branching rates as functions of parameters and number of cones was hard to resolve exactly for the explicit time dependence, conversely a computational method was introduce with great success in reproducing the original model. Some neurite generated with NetGrowth are presented and discussed in Fig.7.5

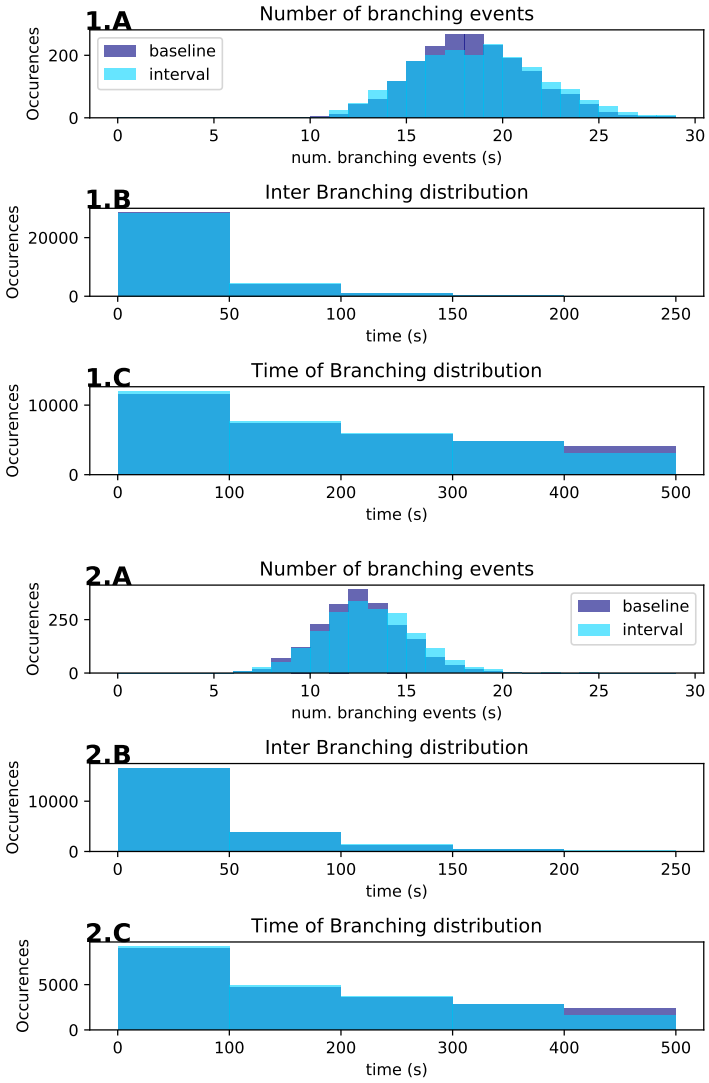


Figure 7.2. Branching times for BEST model Pictures **1** and **2** describe the statistic properties of topological tree generated by the BEST model. The histogram are produced running the algorithm 1000 times. This test verifies whether the approximated method reproduces the results obtained by the BEST model.
Light blue histogram is generated with the fixed interval approximation (see Eq.7.16): the interval time distribution is computed and the branching time is drawn from it. Dark blue one trees are generated with BEST standard algorithm *baseline*: each time step it checks whether or not to branch. In a separate study we have observed the time saved increases linearly with the number of expected branching events. The parameters of baseline $D(t)$ are equal for both the processes: $\tau = 1000$, $A = 1$. The matching of these two methods is tested varying the competition parameter: $E = 0.3$ in **1** and $E = 0.7$ in **2**
The fixed time approximation proposed above fits adequately with the BEST model distribution, pictures **A**. The Inter Branching Interval distribution (IBI) in **B** and average time of branching occurrences are well reproduced too. The approximation is weak for large interval, as explained in Eq. 7.16, and this results appear clearly in the third histogram.

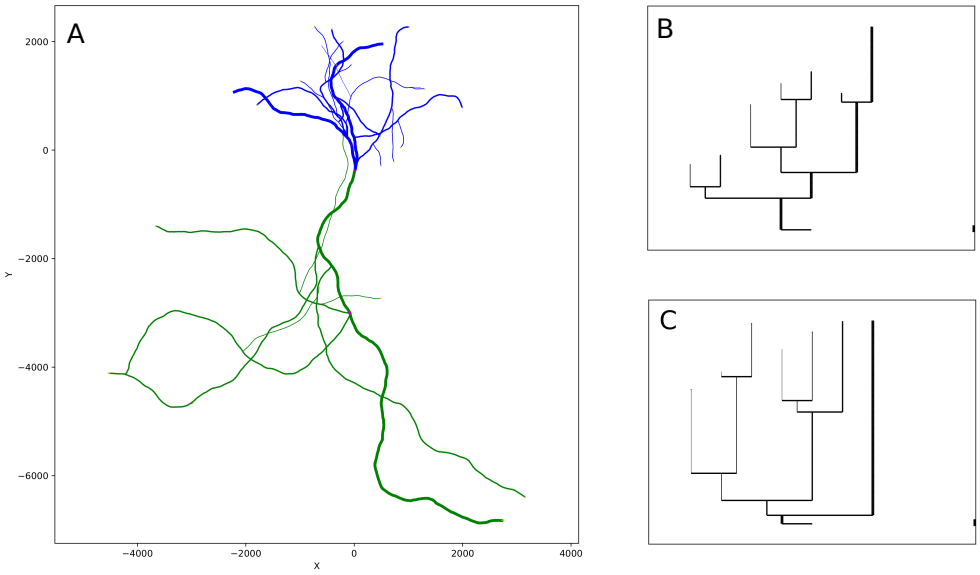


Figure 7.3. Lateral branching, neuron generated with NetGrowth The lateral events can occur along the neurite length whenever an amount of actin erupts from the neurite shaft, in the picture I present a neuron generated with NetGrowth simulator. **A** is the neuron with dendrites in blue and axon in green. It presents only lateral branching: the neurite elongate and then branches with a powerlaw decreasing rate. The neuron is generated with run and tumble method for the GC navigation and the competition among Growth Cones is switched off. As discussed the neuron is generated precomputing the branching time with a fixed rate approximation, this approach saves huge amount of random number drawings.
B and **C** present the dendrogram for other neurites simulated with the same algorithm. In this case is very hard to discuss the tree symmetry and structure: the neurite branches independently by the centrifugal order of the tree. Only the length of the branch matters: the probability of branches decrease exponentially with distance from the leading Growth Cone. Which of neurites is going to ranch is chosen weighting the branch length itself: longer branches have more probability to have an interstitial issue of actin.

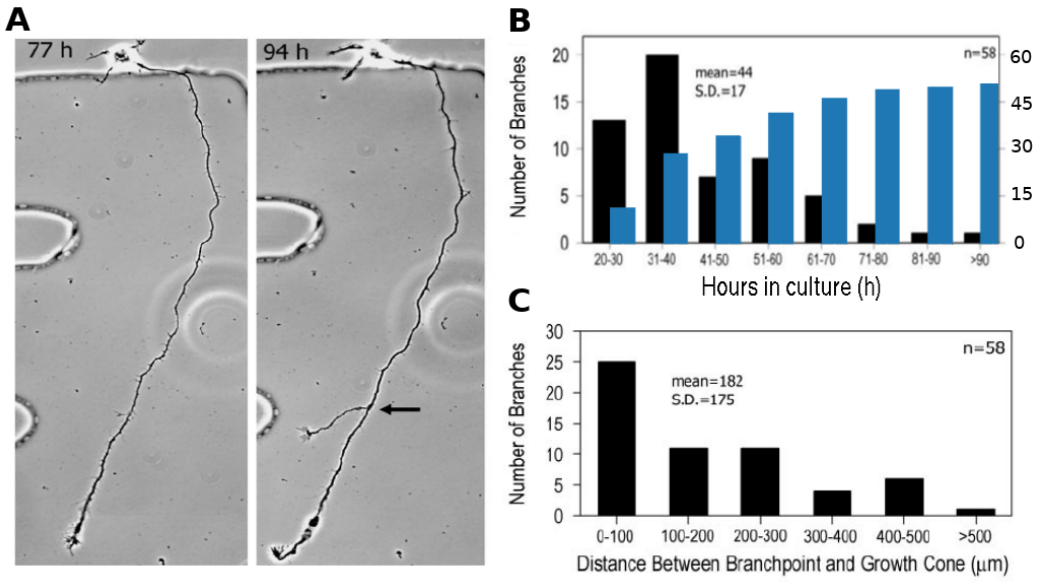


Figure 7.4. Lateral branching, experimental evidences The picture reports two relevant measures from cortical neurons in culture. (**A**) Shows a lateral branching event at 94h from culture initiation. Fig.s (**B**) and (**C**) present a statistics of the observed neurons (58 samples). The former summarizes the frequency histogram, bars show numbers of new branches that extended during each of the time periods indicated. Most of the branches extended on the second and third day in culture.The blue, superimposed, histogram is the total number of interstitial respect to the time periods. The latter is the frequency histogram showing numbers of branches that extended from the axon at varying distances behind the primary growth cone. The functional form and the distribution were used to define the interstitial branching model. Data from [Szebenyi et al., 1998].

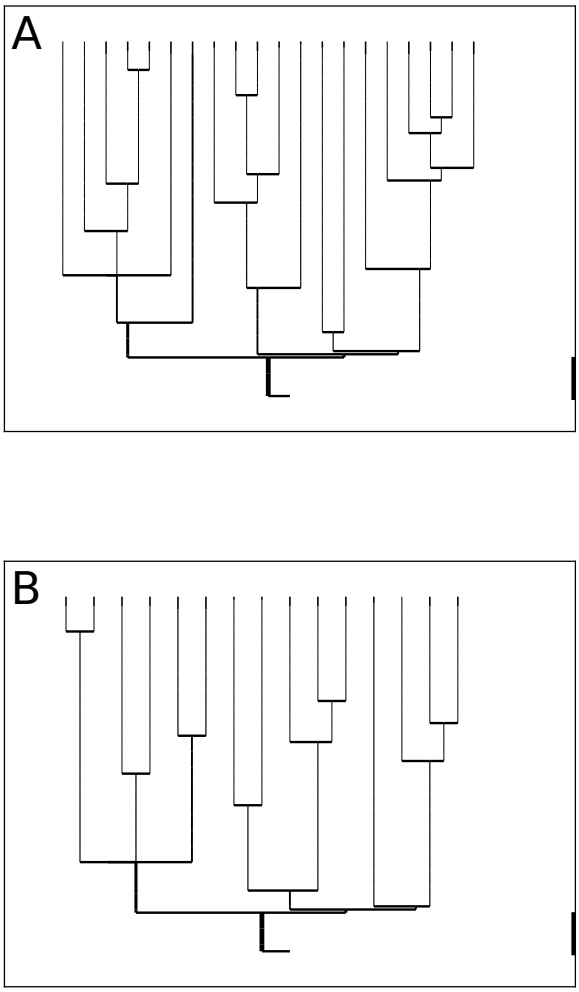


Figure 7.5. Dendritic trees produced with GC split models in NetGrowth The pictures present some of possible dendritic trees realized with the (modified) BEST algorithm. **A** and **B** graphs were produced varying the symmetry parameter: Once the branching time is computed with the fixed rate approximation (see Eq. 7.21) the election of splitting GC is made by the BEST model prescription: the parameter S regulates the importance of centrifugal order, when it is greater then zero the tree is more likely to be symmetric. Also the competition parameter is varied , higher competition prevents cones from splitting. Neurite **A** has $S = 0, E = 0.3$, **B** has $S = 1, E = 0.7$.

7.2 Branching angles and diameter

This short section will focus on the algorithm we implement to produce relevant morphology in the respect of branching angles and diameters. We have followed the data and models proposed in [Shefi et al., 2004]. There are some constraints the splitting growth cones has been observed to respect: First there is a generalized *Rall condition* the branching neurite respects in absence of external cues, the diameter of the splitting growth cones are governed by the simple equation:

$$d_{father} = d_1^{\eta} + d_2^{\eta} \tag{7.22}$$

Second, Shefi &co proposes a model based on forces equilibrium, if the branches are assumed to pull the bifurcation with a certain tension T,hence the relation between diameters and branches angle results:

$$\frac{T_1}{T_2} = \frac{\sin(\alpha_1)}{\sin(\alpha_2)}$$

where α_i is the angle of the branch in respect the parent branch, as depicted in Fig. 7.6. The author proposes the pulling force to be proportional to some power of the diameter, hence the resulting equation becomes:

$$\frac{d_1^{\nu}}{T_2^{\nu}} = \frac{\sin(\alpha_1)}{\sin(\alpha_2)} \tag{7.23}$$

To close the system we have to impose the branching angle $\alpha = \alpha_1 + \alpha_2$, which is derived by experimental observation. The code in 7.1 is the resulting implementation of these simple constraints. We have introduced a noise in Eq. 7.22 to produce variable morphologies, also we have assumed $\nu = 1$ in the absence of other evidences. In case of lateral branching we could not find any report, hence the diameter has been produced with a power law decreasing width.

Listing 7.1. Algorithm for diameter-aware branching

```
\label{code:angles}
// draw the branching angle from a gaussian distribution as
    hypothesized in
// reference article
double branching_angle = gc_split_angle_mean_ +
                         gc_split_angle_std_ *
                             normal_(*(rnd_engine).get());

// ratio between the diameters of the two neurites,
// it's a gaussian distributed value arround 1.
double diameter_variance_ = 0.02; //@TODO here the variance was fixed,
                                  // requires to be set by the user!
double ratio = 1 + diameter_variance_ * normal_(*(rnd_engine).get());
// The diameters are computed on the base of:
// d^eta = d_1^eta + d_2^eta
// next two lines compute the new diameter
// from the old diameter of the neurite and the ratio between the
    rising
```

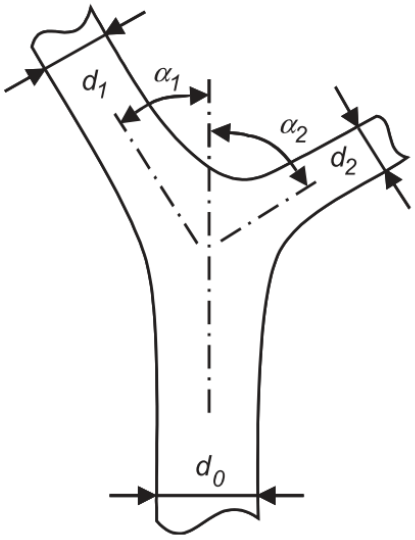


Figure 7.6. Angles and diameters, a mechanical equilibrium model Following the experimental evidences reported in [Shefi et al., 2004] I implemented a simple algorithm 7.1 to reproduce the mechanical equilibrium model proposed in 7.23,7.22. With this model the branches pull the bifurcation with a strength proportional to their diameters. This hypothesis is in agreement with the focal adhesion model, which predicts the pulling action is exerted over the substrate by the molecular motors.

```
// cones.
new_diameter = old_diameter * powf(1. + powf(ratio, diameter_eta_exp_),
                                   -1. / diameter_eta_exp_);
old_diameter = powf(powf(old_diameter, diameter_eta_exp_) -
                    powf(new_diameter, diameter_eta_exp_),
                  1. / diameter_eta_exp_);

// the diameter affect the branching angle: the largest cone goes
    straighter
// than the other.
double eps = (old_diameter - new_diameter) / (old_diameter +
    new_diameter);
double half_tang = eps * tan(branching_angle / 2);
new_angle      = old_angle;

new_angle = -branching_angle / 2. - half_tang;
old_angle = +branching_angle / 2. - half_tang;
```

7.3 Competition for critical resource

In chapter 5.4.1 we have introduced a simple model for critical resources competition among growth cones. This model is based on the hypothesis that GC might require a specific molecule to grow, which would be available through a given amount throughout the neurite. In a complex neurite the GCs compete for this resource

and their dynamics (elongation, retraction, stalling or splitting) depends on the amount of resource that is available to them. In the following chapter some results are obtained for a simplified model. Our model avoids the integration of the diffusive equation over the whole neurite; conversely, it follows the solution of a system of SDE, one for each GC. I have to underline this model is the last of a vast amount of competition models, defined and studied during my period in Paris. I will describe the previous attempts in the Appendix D to avoid an useless complication of the manuscript. We will proceed introducing the ideas behind and then relating them to the equation. The stochastic differential equations are managed by the formalism used in [Boffetta Guido, 2012] [e Parisi, 2010]

7.3.1 Strive to grow: from ideas to formal models

The ideas shaping this model are borrowed by [Ooyen, 2004]. The fundamental principle is the amount ot the critical resource received is necessary to synthesise microtubules or to keep them together [Hely, 2001]. Hence the amount of critical resource sets the elongation rate and, possibly, the branching rate too.
The first models tested were equivalent to the stochastic distribution of a quantity over the leaves of a tree. In this frame we had introduced the geometrical factor ζ_i, it reduces the received amount for high order branches. The received resource cannot be completely consumed because some other molecules are necessary: these elements can be produced in the GC itself with a certain rate u. Same how the resource corrupts with a rate τ_l. The consumption rate should not be fixed but depend by externally triggered or stochastic variations, χ. Thus the distribution follows the consumption; this picture of the process motivates the second term in r.h.s of Eq. 7.24. Eventually the critical resource produced in the soma can fluctuate with its own timescale, imposing the second equation in Eq.7.24. Starting from this sketch, for a growth cone i, we get:

$$\begin{cases} \dot{a_i} &= -a_i \left(\underbrace{u + \frac{1}{\tau_l}}_{\kappa} \right) (1 + \chi_i) + \frac{A}{\tau_d} \frac{\zeta_i \kappa a_i}{\sum \zeta_j \kappa a_j} \\ \dot{A} &= \frac{A_m - A}{\tau_A} - \frac{A}{\tau_d} + \xi \quad = \quad \frac{1}{\tau}(A_M - A) + \xi \end{cases} \tag{7.24}$$

where $\xi \in \mathcal{N}(0, \sigma_\xi, \rho)$ and $\chi_i \in \mathcal{N}(0, \sigma_i, \rho_i)$ are potentially correlated Gaussian random variables. In these equations:

a_i is the quantity of resource available,

u is the consumption rate,

τ_l is the leak timescale,

ζ_i is the weight factor (geometrical) of the GC,

A is the amount of resource available inside the neurite and which can be delivered to the GCs,

τ is such that $\tau^{-1} = \tau_A^{-1} + \tau_d^{-1}$

$A_M = \tau_A^{-1} \tau A_m$

Once the quantity of critical resource is obtained, we can define the dynamics of the growth cone's speed v based on the thresholds:

θ_{rs} below which the GC retracts

θ_{se} above which the GC elongates

such that:

$$v_i = \begin{cases} \frac{a_i u - \theta_{rs}}{\theta_{rs}} v_r < 0 & \text{if} \quad a_i u < \theta_{rs} \\ 0 & \text{if} \quad \theta_{rs} \leq a_i u \leq \theta_{se} \\ \frac{a_i u - \theta_{se}}{A u - \theta_{se}} v_e > 0 & \text{if} \quad \theta_{se} < a_i u \end{cases} \tag{7.25}$$

where:

v_r is the maximum retraction speed when a GC gets 0 resource

v_e is the maximum elongation speed when a GC gets all the resource generated by the soma

Eventually we also define the threshold θ_S, above which the GC can split.

7.3.2 Solutions for the amount of generated resource

Before implementing the models we wanted to forecast its average behaviour and the dynamics that can arise by it. Some of the used mathematical tools are reported in the Appendix C. The stationary distribution for the amount of critical resource produced is:

$$f(A) = C \exp\left[-\frac{(A - A_M)^2}{2\sigma_f^2}\right] \tag{7.26}$$

And the solution of the average variable $\langle A \rangle$ is given by:

$$\langle A \rangle(t) = A_M + (A_0 - A_M) e^{-t/\tau} \tag{7.27}$$

Which has been represented in Fig. 7.7, in respect to elongation and retraction threshold.

7.3.3 Solution for one growth cone (white noise)

Let's move our attention to the dynamics of the growth cones, assumed the equations 7.24 hold for a system with N growth cones the case for $N = 1$ follows:

$$\begin{cases} \dot{a} &= -a\kappa\,(1 + \chi) + \dfrac{A}{\tau_d} \\ \dot{A} &= \dfrac{1}{\tau}(A_M - A) + \xi \end{cases} \tag{7.28}$$

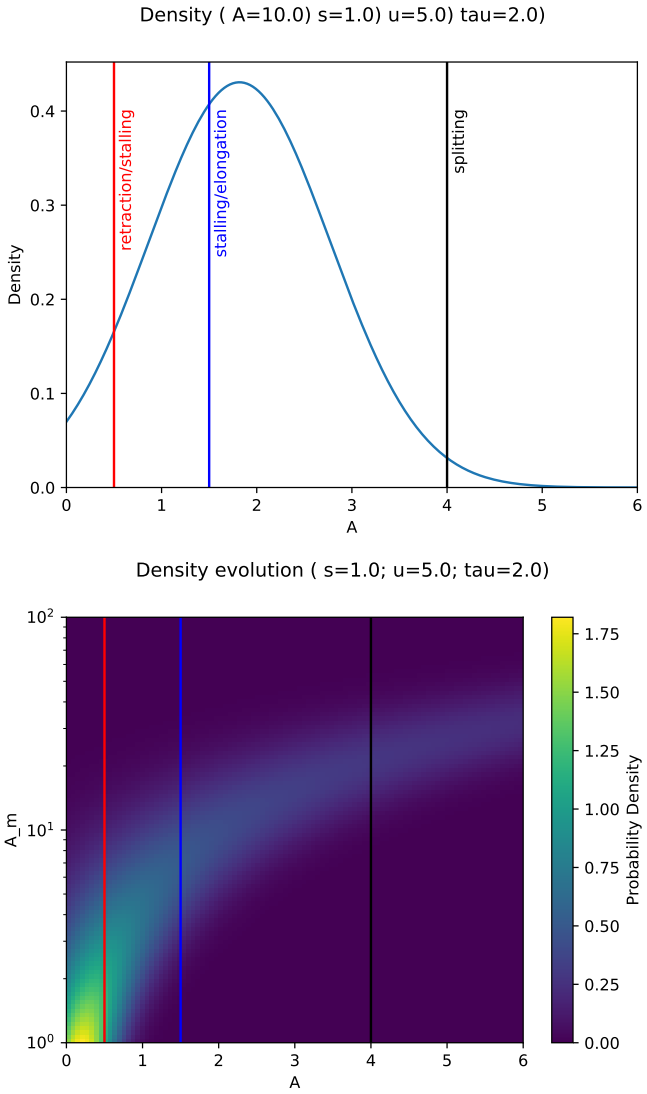


Figure 7.7. Single Growth Cone with critical resource picture describes the stationary probability distribution for a Growth Cone in the critical resource model, the vertical lines indicate the thresholds for retracting or elongate. The models parameter can set the mean above or below the threshold, intuitively we expect for an increasing number of Growth Cones the mean rate of received resource falls below the elongation threshold, further studies should be performed in this direction, studying the distribution for a fixed number of growth cones, and eventually solve the model for an increasing number of Growth Cones.

This lead to the average and squared average:

$$\langle \Delta a \rangle = -\kappa \langle a \rangle \Delta t + \frac{A}{\tau_d} \Delta t \tag{7.29}$$

$$\langle (\Delta a)^2 \rangle = a^2 \kappa^2 \sigma^2 \Delta t + o(\Delta t) \tag{7.30}$$

which differs from a Ornstein-Uhlenbeck process for the presence of a as a factor of the stochastic component. Considering the average equation for A and assuming (if $\kappa \neq \tau^{-1}$) we can expand 7.28

$$\langle a \rangle(t) = \frac{A_M}{\kappa \tau_d} + \langle a \rangle_0 e^{-\kappa t} + \frac{\tau}{\tau_d(\tau\kappa - 1)}(A_0 - A_m)e^{-t/\tau} \tag{7.31}$$

or to

$$\langle a \rangle(t) = \frac{A_M}{\kappa \tau_d} + \langle a \rangle_0 e^{-\kappa t} + \frac{A_0 - A_M}{\tau_d} t e^{-\kappa t} \tag{7.32}$$

if they are (we do not consider this case since we will consider $\kappa \gg \tau_A^{-1}, \tau_d^{-1}$).

Conditional Fokker-Planck

Let $g_A(a,t) = f(a,t|A)$ be the conditional distribution for a at fixed A.
We get:

$$\partial_t g_A = (\sigma_\chi^2 - \kappa) f + \left(2a\sigma_\chi^2 + \frac{A}{\tau_d} - \kappa a\right) \partial_a g_A + \frac{(a\kappa\sigma_\chi)^2}{2} \partial_a^2 g_A \tag{7.33}$$

$$= \partial_a \left[\left(\frac{A}{\tau_d} - \kappa a \right) g_A + \partial_a \left(\frac{(a\kappa\sigma_\chi)^2}{2} g_A \right) \right] \tag{7.34}$$

$$= \partial_a \left[\left(\frac{A}{\tau_d} + a\kappa(\kappa\sigma_\chi^2 - 1) \right) g_A + \frac{(a\kappa\sigma_\chi)^2}{2} \partial_a g_A \right] \tag{7.35}$$

with, since $a \geq 0$,

$$\forall t,\ \partial_a g_A(0,t) = 0. \tag{7.36}$$

7.3.4 Competition between two GCs

Let there be two GCs, denoted be the subscripts 1 and 2, the evolution of their available resource in time is given by:

$$\begin{cases} \dot{a}_1 = -\kappa a_1(1 + \chi_1) + \dfrac{A}{\tau_d} \dfrac{a_1}{a_1 + a_2} \\ \dot{a}_2 = -\kappa a_2(1 + \chi_2) + \dfrac{A}{\tau_d} \dfrac{a_2}{a_1 + a_2} \end{cases} \tag{7.37}$$

where $\chi_i \in \mathcal{N}(1, \sigma_i, \rho_i)$. This system can also be describe by the symmetric and antisymmetric components:

$$\begin{cases} \dot{\alpha} = -\kappa\alpha + \dfrac{A}{\tau_d} - \kappa \sum_i a_i \chi_i \\ \dot{\beta} = \left(\dfrac{A}{\alpha \tau_d} - \kappa \right) \beta - \kappa \left(a_1 \chi_1 - a_2 \chi_2 \right) \end{cases} \tag{7.38}$$

with $\alpha = a_1 + a_2$ and $\beta = a_1 - a_2$. The equation is integrated numerically and the results are presented in Fig. 7.8. The method chosen for the integration is the Heun method, also known as Improved Euler Method, the method is recall in Appendix V. The algorithm has been tested in a test suite, here we have verified the results were consistent, i.e. the growth cones where advancing or retracting by the amount of received critical resource, and there was actually a competition. Once the model was considere effective we have implemented it to the simulator, the weigth ζ_i was expanded into two weigths, one topological and one for the diameter, in the idea the larger the neurite, the most the received resource. In Fig. 7.8 the case of two or three competing GCs is discussed.

7.3.5 More than two Growth Cones

In case the neurite starts to branch repeatedly, which is normally the case, the result becomes soon unpredictable, the space of parameters of the model is broad, and whether we hope they can be biologically related with a research project focused on it, for the moment we have to guess the right set of parameters to produce relevant and plausible neurons. In the following pages three case will be evaluated: firstly the effect of an unbalance between the outgrowing cones, this inhomogeneity is driven by the neurite diameter in this case, Fig. 7.9. Second the relevance on the final morphology by the amplitude of stalling regime, among the retracting and elongating thresholds Fig. 7.10. Eventually the morphology obtained changing the correlation of the stochastic variable ξ, in case of a large variance, is reported in Fig 7.11.

7.3.6 Conclusion

In the first section I have performed a study over the branching rate of BEST model to adapt it to the efficiency requirements of NetGrowth, I have both assessed analytical arguments and empirically verified the new method reproduces the topologies generated with the BEST model. The lateral (interstitial) branching was approached with the same method but with a different distribution, I have traced the regime of validity of the approximation too. Hence I reported the algorithm implemented in NetGrowth to regulate the branching angles and the diameters of sibling neurites. Eventually the elongation rate of Growth Cones has been modeled with an original competition model, it was devised basing on previously discussed evidences [van Pelt et al., 2003b], [Hely, 2001]. After an intense effort of modelling, discussed in Appendix D, a system of two SDE has been devised to solve the problem. The differential equations governing the model have been analytically resolved with FP method, where possible, and integrated with the Heun method (Appendix V) when not.
With this model the neurite is simulated as a complex structure, where the relative outgrowth of some components affect negatively the elongation of other growth cones. Some experiments can be devised to test whether this approach is meaningful and relevant, these experiment can be arranged with neurons deprived of tubulin or whose MAP2 protein generation is inhibited. In this case the rate of resource produced and diffused can be related to the parameters of the model and test which conditions can be reproduced with NetGrowth and which not.

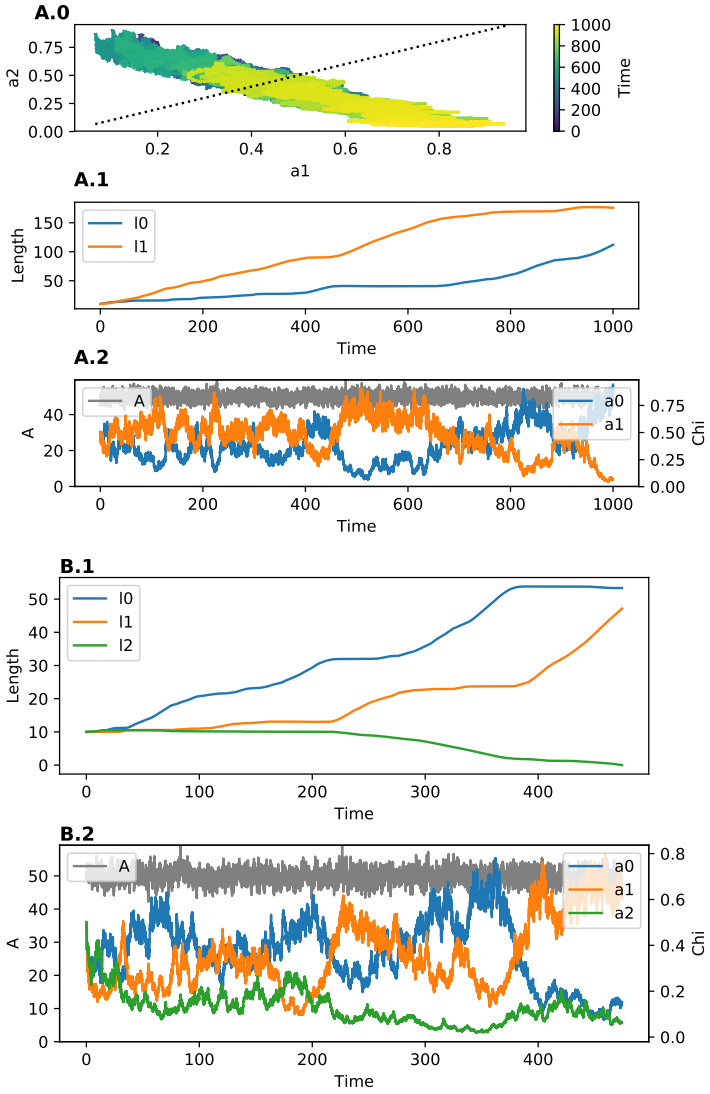


Figure 7.8. Base study for competition model Here it is reported the separated study performed for the competition model. The equations in 7.24 are implemented with the Heun method for two **A** and three **B** growth cones. This test suite helped into study the overall effect of competition on outgrowing cones. Picture **A.1** report the amount critical resource obtained during the simulation, by each of the GC. The amount of resource accessed from one cone is inversely proportional to the one accessed by the other, the width is due to the fluctuations of the generated resource, color proxies for time evolution.

Pictures **A.2 and B.1** indicate the distance from the soma for the growth cones as a function of time, the position changes during the time and gcs elongate or retract depending by the critical resource received. Eventually pictures **A.3, B.2** show the amount of CR received, obviously it is symmetrical for two growth cones and less trivial for three, the grey line is the critical resource generated in the soma.

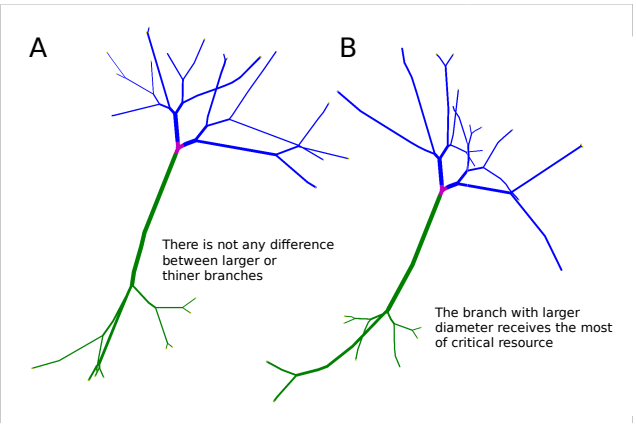


Figure 7.9. Competition enhanced by neurite diameter In the picture two neuron growth with the critical resource model are presented. The neurons are produced with same parameters except for the weight ζ_d which triggers the competition by diameter. In the neuron **A** all the branches receive, approximately, the same amount of critical resource, which is consumed independently by the neurite size. In the neuron **B**, instead, the flux of substance received depends strongly by the diameter, this dependence prevents shorter branches to elongate and promote larger ones. This mechanism can be further enhancedwith a topological dependent weight, it will forces the tree to grow symmetrically or not. The weight becomes more relevant when the pruning process is taken into account, in this case the weaker branches not only stay behind but get pruned on the long run.

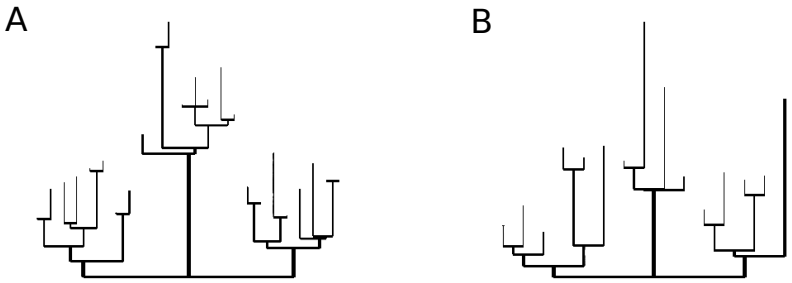


Figure 7.10. Competition with low or high elongation threshold This picture presents the dendrograms of two neurons growth with different parameters in respect to the elongation and retraction thresholds θ_{el}, θ_{re}, both the case are diameter independent. The dendrogram on the left **A** present some pausing growth cones, i.e. the plate branches, while the one on the right **B** has only outgrowing branches. The latter dendrogram presents, indeed, a greater value for the elongation threshold, so the stalling regime is reduced and the inactive growth cones get pruned, leaving more substances to longer growth cones. This configuration can be particularly relevant for the axon, which can choose the direction following an evanescent gradient of chemiotaxic field and set all the neurite effort to reach the target.

In the next chapter an overall estimation of the NetGrowth simulated neurons is offered, highlighting strengths and flaws of the project.

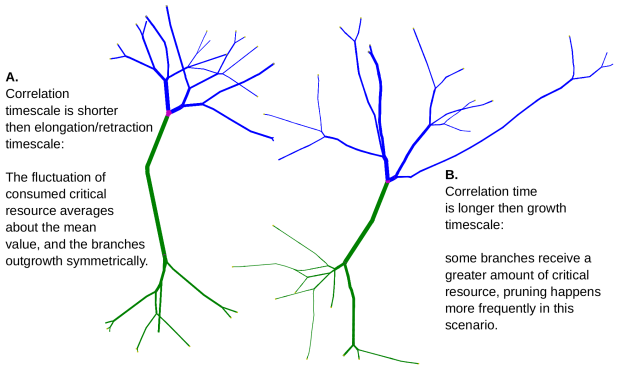


Figure 7.11. External stimuli with different timescales This example shows the effect of randomness when related to the consumed critical resource. The idea is: there are fluctuations in the employment, or in the leakage of CR. these fluctuations are due to internal or external factors and have a certain intensity and time scale. The intensity can be set by the variance of the stochastic variable ξ while the time scale can be regulated with the correlation factor. In the example two neurons -growth with same parameters and different correlation times- present characteristic behaviours: in the first case **A** the neuron is quite symmetric on all its branches, indeed the stochastic variable is quickly averaged about its mean value (i.e. zero) and neither retraction or elongation is enhanced. In the second case **B** some branches are stretched way more than their siblings, we expect also more pruned branches in this case.

Part III

Evaluation of NetGrowth simulator

Chapter 8

From neurons to networks, validation and graph creation

Contents

8.1 Validating Netgrowth morphologies

The validation of NetGrowth simulator is a fundamental part of the project: the morphologies and networks generated should be a close reproduction of those observed in neurobiology laboratories. Some test cases on which NetGrowth could be used are the following:

- **How does the inter population connectivity change when the average branching rate is increased?** To aid neurobiologist in relating the macro-scale properties of single neurons and neuronal networks on Petri dish, i.e. connectivity, graph closeness measures, arbor extension, etc. . . , to the micro-scale processes and parameters: The modularity of the simulator allow for the elaboration and implementation of new models in a parallelized simulator, and we hope once the simulator will be ready to use some labs will pay attention to its possibilities.

- **What is the resulting network of a certain culture structure? How do topologies affect synaptic plasticity?** To furnish plausible networks and morphologies to theoretical researchers fuels the investigation of more plausible network topologies, which, integrated into activity simulator, e.g. NEST, allow for the simulation of cultured neurons dynamics. The verisimilitude is a necessary requirements to keep the investigations relevant and prone to applications.

Having stressed the importance of obtaining realistic neuron morphologies and effective environment interactions, then comes the hardest and trickiest part of the whole project: testing the verisimilitude of the generated neurons and networks. As generally done in science the *dividi et impera* approach should make the case, indeed the properties to test are many and verifying the branching rate together with the elongation model could not unveil the insufficiencies of one or the other formal descriptions. The simulator should be confirmed by appositely conceived experiments, as those used to benchmark the environment interactions. These experiments require time and competence, and were not performed during the short time of my master thesis.

On the other hand one of severe lessons I learnt during these project is what happens inside a cell is never obvious: The chemo-electrical processes going on beyond cellular membrane are tightly connected, and to study them separately is not an easy achievement. It is possible, for this purpose, to inhibit some mechanisms, as the branching, with drugs or genetic manipulation, but it requires an entire research project and often offers only qualitative results. Furthermore the behaviour of the single neuron depends thoroughly by the environment, i.e. the rest of the culture. Neurons are not solitary, either their lifespan is strongly correlated to their environment and to the presence of other cells. Glia, for instance, are essential to feed the neurons and form synaptic connections, same how the neuron will not survive if they don't establish synaptic connections, and when they do they prune large part of their body which is not functional to its scope: receive and transmit electrochemical signals. With these premises to test the NetGrowth simulator would require a long session of experiments, the parameters should be fine tuned with observation and accurately related to live cells behaviour; different environment should be testes and see whether they affect or not the growth and how. This was not possible in the context of my thesis, therefore only a short hint of validation will be presented in this manuscript.

The validation will be divided in two sections, in the first I will review two articles which offer a consistent blueprint of measures, towards an integrated and effective statistic of neuron morphology. In the second I will produce these measures for some morphologies generated with NetGrowth.

8.1.1 Measuring neuron's morphologies

The articles we are referring for morphologies validation are [Costa et al., 2010, Snider et al., 2010]. The authors of these review concentrated on general properties of neurons' morphology, stressing the relevant aspects of neurites, Fig 8.1 and the consequent measures can be performed to classify these neurons and to validate *in silico* generated neurons. These approach unveils the most of neurons in a reduced area of the 2D space, which is very reduced in respect to the neuron that can be generated *in silico* both with generative and statistical procedures. Anyway a relevant properties of neuronal morphology is the spatial density function, a function that specifies the probability of finding a branch of a particular arbor at each point of the real space. From the analysis of over a thousand arbors from many neuron

types in various species, [Snider et al., 2010] has discovered an unexpected simplicity in arbor structure: all of the arbors examined, both axonal and dendritic, can be described by a Gaussian density function truncated at about two standard deviations. These result, very surprising indeed, is valid for neuron growth with a completely isotropic space and can be very useful to verify the quality of morphologies produced with NetGrowth. In respect to the previous analysis the classification is reduced to four parameters in 3D and three in 2D: the total length of dendritic tree and the 3(2) parameters for the standard deviation in each dimension, an example is reported in Fig. 8.2. Another standard measure can is the Sholl Analysis: a method of quantitative analysis commonly used in neuronal studies to characterize the morphological characteristics of an imaged neuron [Sholl, 1953]. It was first used to describe the differences in the visual and motor cortices of cats. While methods for estimating number of cells have vastly improved since 1953, the method Sholl used to record the number of intersections at various distances from the cell body is still in use and is actually named after Sholl. In order to study the way in which the number of branches varies with the distance from soma, it is convenient to use a series of concentric spherical shells as co-ordinates of reference these shells have their common centre in the soma. Sholl also realized his method is useful to determine where and how big is the region where synapses are possible, an example of Sholl analysis on a real neuron is offered in Fig. 8.2

In the next section the morphologies produce with NetGrowth in isotropic space will be validated with the method of arbor density.

8.1.2 Simulated and cultured neurons

This section is indeed very relevant for the scientific validity of the work I have done for the NetGrowth simulator. Since the simulator will produce morphology for confined culture a sound validation of the morphologies should be done in that direction. To studies these morphologies *in vitro* it is, sadly, very hard. The arising complications are many, one over all the death of the selected neuron. Furthermore the acquisition of the image into a descriptive format is not trivial, many software exist to do it and the most are proprietary and sold with the microscope itself. Eventually we have to consider the evaluation of the morphology cannot be done on general basis, indeed the morphology depend deeply by the parameters of the experiment: the area of the brain whose cells belong, the species of the animal, the conditions of growth and the time of the neuron. It is evident that a sound work of validation should be done with the help of a neurobiologist and with the facility of a neurobiology laboratory. These accomplishments are well beyond the possibilities, in terms of time and knowledge, of this manuscript and are left open. Anyway the measures themself will be performed on the produced neurons and attached, both to prove the results are similar to those we can find on publications and in the web in general, and to figure out how the conclusion of the NetGrowth project will appear. Even to find neurons in 2D in the web is a hard task, indeed the NeuroMorpho database collects only neurons obtained from in vivo, and it lacks of neurons growth on Petri dishes. Since the following analysis is completely arbitrary I opted to pay tribute to Santiago Ramón y Cajal, the first neuroanatomist. He was fascinated by the complexity of neurite arbor and draw a huge quantity of hand-made reproductions

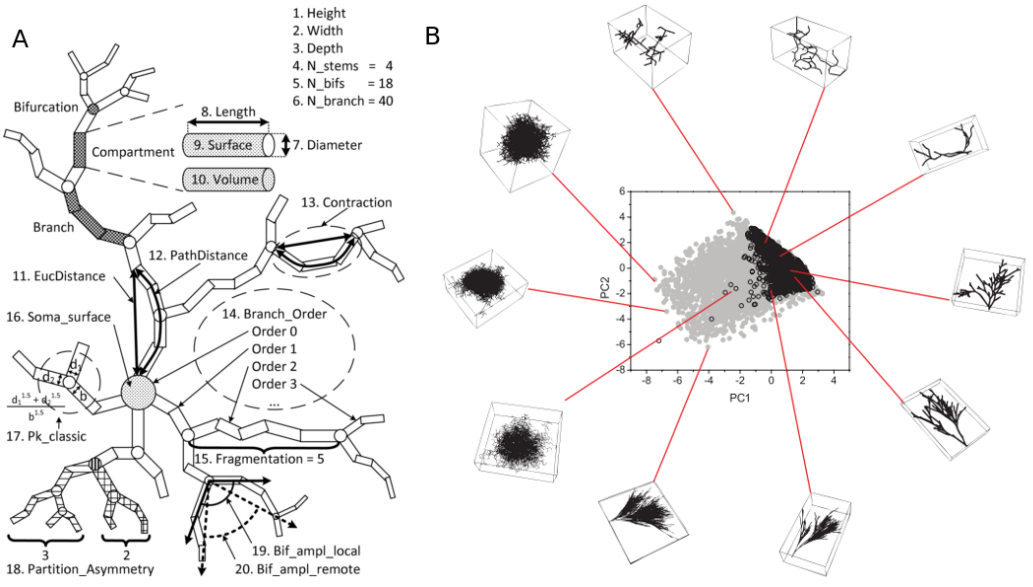


Figure 8.1. The variables of neuromorphology space The neurites are complex structures and their geometry and topology can be classified on the base of 20 parameters, picture **A**. The variables can be global, as the number of total branches, or local, (i.e. the distance between two branching points, the angle between two sibling branches, etc.... The author proposes a PCA description, which maps the many variables in a 2D space, picture **B** which is more easy to visualize and compare, in the picture the grey spots are the morphologies produced in *in silico* while the black spots are those obtained by the NeuroMorpho database. [Costa et al., 2010]

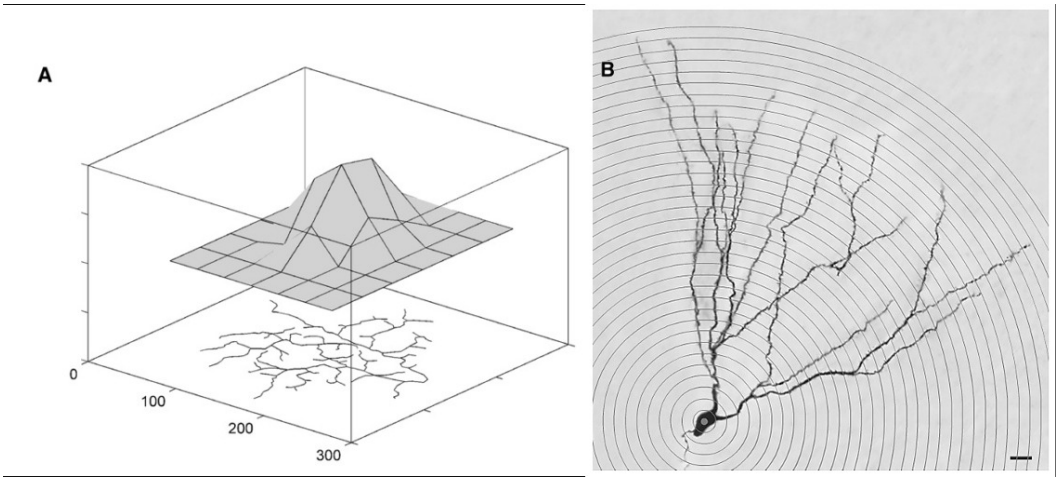


Figure 8.2. Density of dendritic arbor and Sholl analysis A broad study, on thousands of reconstructed morphology, shows the neurite has a density function which is Gaussian isotropic space, picture **A**, and proposes that four parameters are enough to describe the dendritic arbor, the dendritic arbor length and the standard deviation in each direction.[Snider et al., 2010] A similar analysis, picture **B**, is used since 1953 to classify neuron morphologies, its inventor, which gave the name to the method, asserted Sholl analysis is valid to estimate the probability of creating a synaptic bottom at a certain distance from the soma.[Sholl, 1953]

of neurons from *in vivo*, his work was the basis for dendritic arbor studies for half of a century. Hence I based on his picture in Fig. 8.3 to draw, *by keyboard* some plausible morphologies. The morphologies I chase are the Chandelier interneuron in Fig. 8.4 and Multipolar cell Fig 8.5. I tuned the parameter by hand, basing on the experience, and the default parameters obtained from articles and review, to reproduce these morphologies. The description of the parameters is reported in the picture.

8.2 NetGrowth simulated cultures

The scope of NetGrowth is not only to produce valid isolated neurons but to create complex networks simulating the growth of neurons in complex confinements. In this section two example of such ability will be offered, the first is a very classic configuration, with hundreds of neurons contained into a Petri dishes. The second example is derived instead from the set of experiments realized by R. Renaud at IPGG, in this case the neurons will be confined in a two-chambers culture. The number of neurons is usually around hundreds of thousands in these cultures, but the simulations were performed on a personal computer, whose limited RAM and CPU does not allow for such a huge simulation. The analysis of the networks obtained is beyond the scope of these thesis: the scenarios which are opened with this simulation are numerous and wide. For instance the degree distribution can be investigated and compared with *in vitro* measured degree distribution. Hence it is possible to compute the centrality of neurons, and see whether their inhibition slows down the activity of the whole culture.
The very important step is to connect the neurons each other throughout the synapses; these chemoelectrical buttons are among the most amazing and complicated elements of neuronal dynamics studies: they can be excitatory or inhibitory and the understanding of the most complex functions of nervous system pass by them. The general mechanism is the following: the pre-synaptic neuron releases chemical elements into the synapse, these are absorbed by the post-synaptic and activate or inhibit the ion channels, favouring or preventing the rise of dendritic spikes. Synapses are not considered now, in next sections the neurons will be considered connected whether the intersection among the dendritic tree and the axon's branches is not empty.

8.3 Conclusions

In this chapter I have introduced some general measures that can help in validating the morphologies produced with NetGrowth. The validation of the produced neurons is a required step to introduce our simulator in the scientific community. Indeed, a severe control is necessary before use NetGrowth as a trustable tool, it will not be possible to commend the simulator to produce networks if we are not sure it can produces realistic neurons.
Nonetheless the importance of the validation phase I was not able to fulfil a sound and rigorous characterization in the framework of my thesis. To mend this fault I generated and discussed some neurons, following the pictures of Ramon y Cajal.

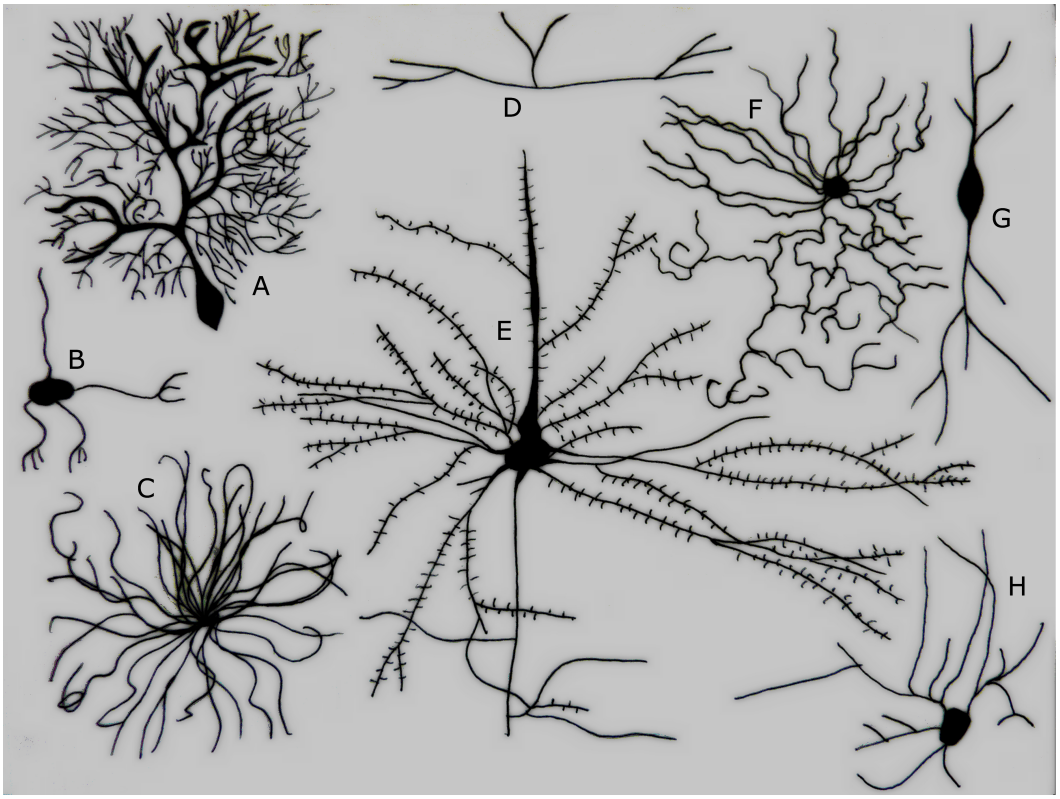


Figure 8.3. Different Types of Neurons drawn by S. Ramón y Cajal A. Purkinje cell **B**. Granule cell **C**. Motor neuron **D**. Tripolar neuron **E**. Pyramidal Cell **F**. Chandelier cell **G**. Spindle neuron **H**. Stellate cell, based on reconstructions and drawings by Cajal. Ramon y Cajal was a pioneer investigator of neurite morphologies and function, he has gained from is observation some fundamental rules that are still in force, e.g. the law of cytoplasm conservation.

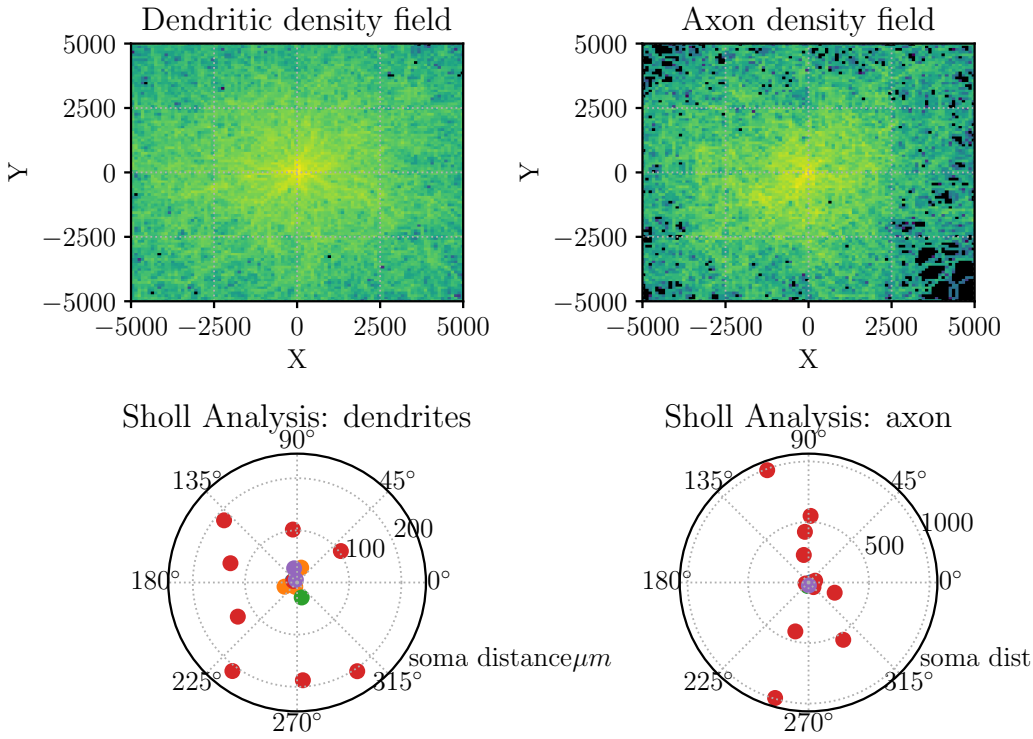


Figure 8.4. Chandelier neurons or chandelier cells are a subset of GABA-ergic cortical interneurons. Chandelier neurons synapse exclusively to the axon initial segment of pyramidal neurons, near the site where action potential is generated. It is believed that they provide inhibitory input to the pyramidal neurons, but there are data showing that in some circumstances the GABA from chandelier neurons could be excitatory.
The neuron is generated gathering the all the model at different time: initially the axon is left to elongate, while the dendrite suddenly undergoes growth cone splitting. Hence the dendrites complete the branching phase and start to elongate in a symmetric competition phase. The axon instead starts to branch, both laterally and growth cone splitting, the persistence length is set about $40\mu m$ but the turning angle is broad $0.5rad$, this produce a characteristic *broken line* aspect to the neurite. In the four plots are presented the Sholl analysis and density analysis for a set of 400 generated neurons. The colors in Sholl analysis distinguish different neurons.

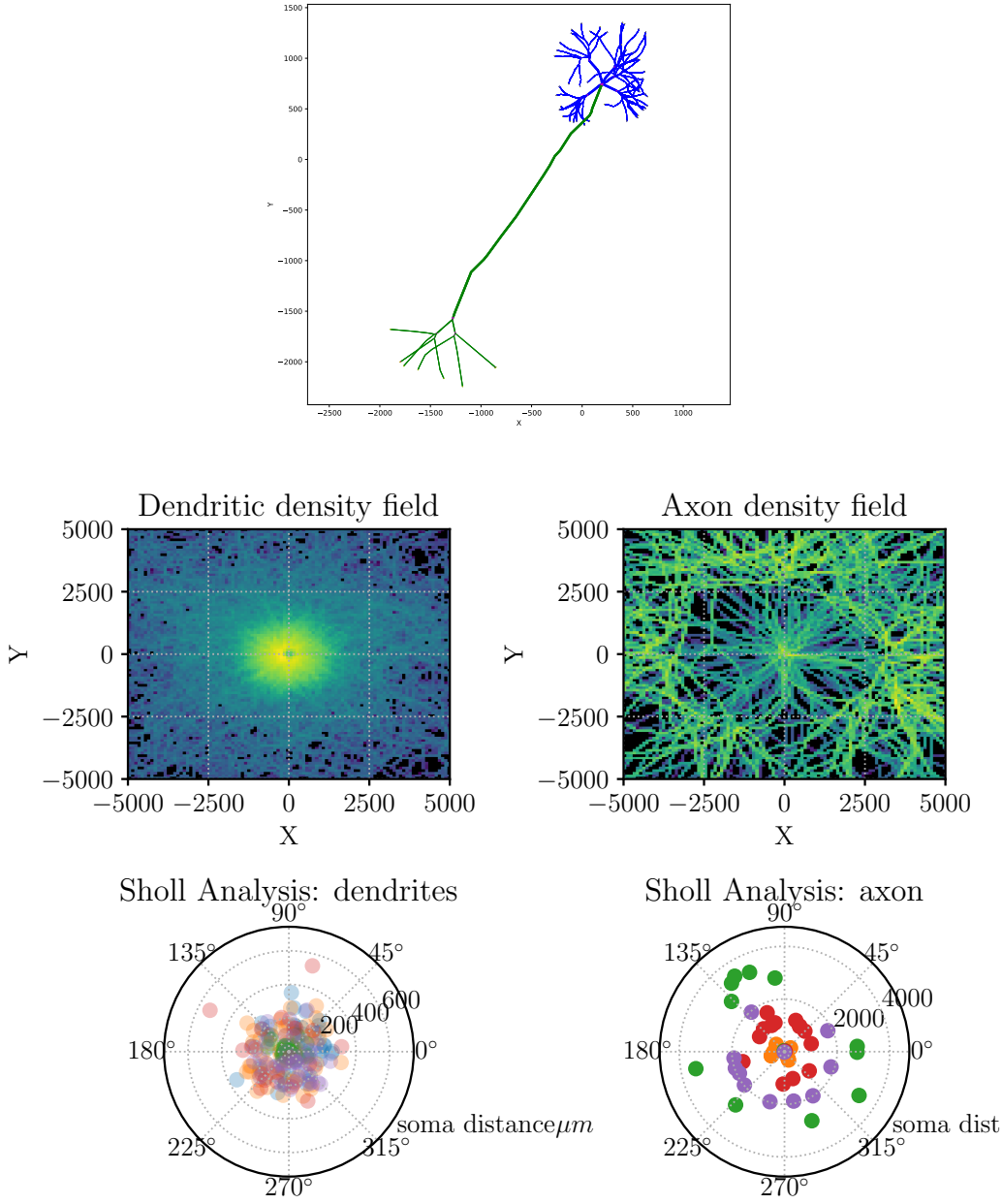


Figure 8.5. Multipolar neurons A multipolar neuron is a type of neuron that possesses a single axon and many dendrites (and dendritic branches), allowing for the integration of a great deal of information from other neurons. These processes are projections from the nerve cell body. Multipolar neurons constitute the majority of neurons in the central nervous system. They include motor neurons and interneurons and are found mostly in the cortex of the brain, the spinal cord, and also in the autonomic ganglia.
The neuron is generated regulating the growth cone splitting to start at different times. The axon elongates rapidly with a strong persistence length, while the neurite branches and compete each other. Hence the growth cone splitting is turned on and the axon starts to branch. The other pictures present the Sholl analysis and the density profile for a set of 400 neurons, while the density of dendrites is tight to the soma, the axons cover a large ring, obviously the diameter of the ring can be tuned by parameters.

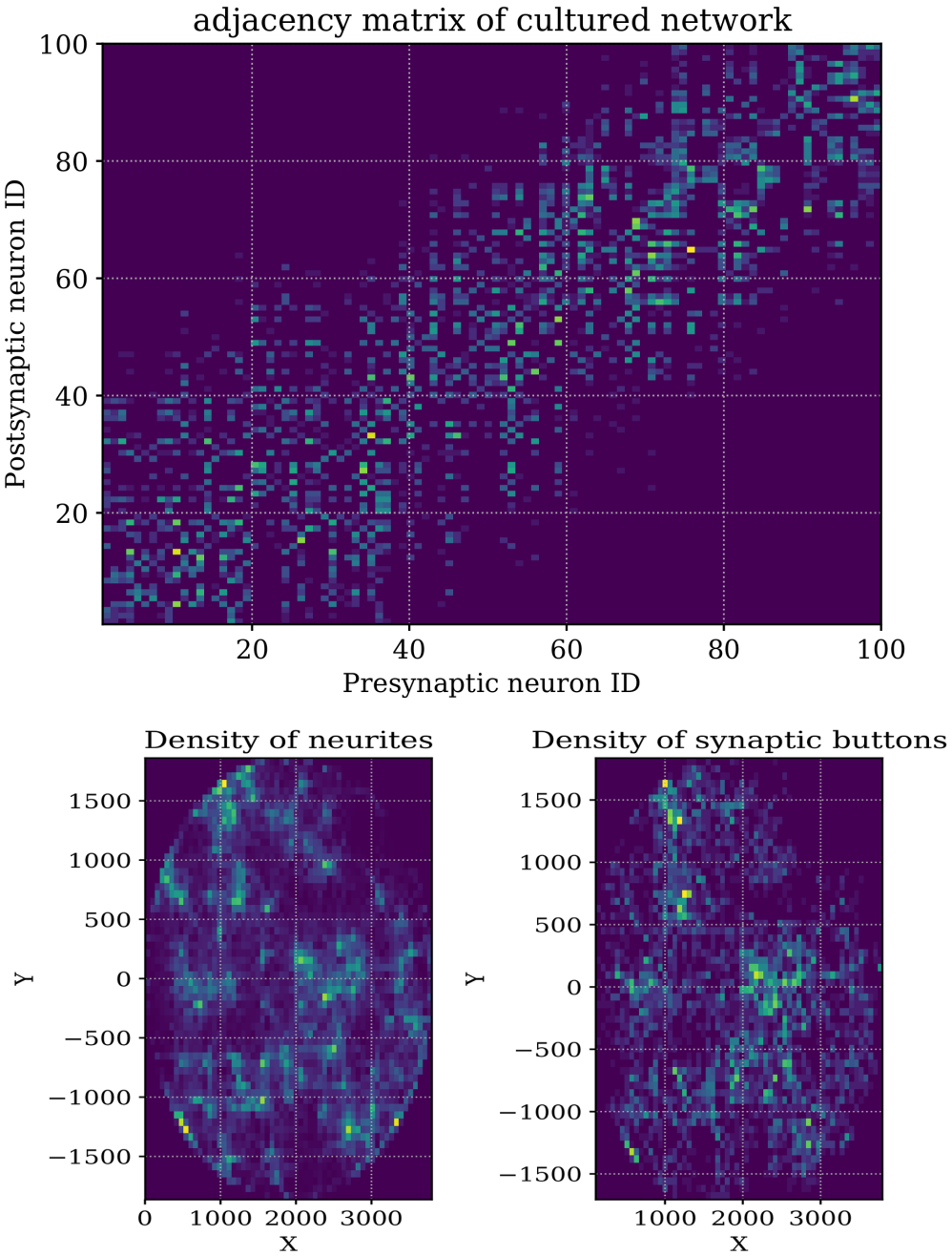


Figure 8.6. NetGrowth generated neurons in Petri dish culture The figure presents results from a Petri dish like confinement with 100 neurons.The neurons are growth within the environment with BEST model branching model and run and tumble navigation model. The data produced are saved in SWC format and post processed with tools I developed within NetGrowth project. Picture **A** reproduces the adjacency matrix obtained from neurite intersection method: whether an axon intersects the dendrite of another neuron a synapse is generated. Pictures **A.1** and **A.2** describes the density of neurites, both axon and dendrites and the occurred synapses. The correlation is evident, this is due to the triviality of intersection method.

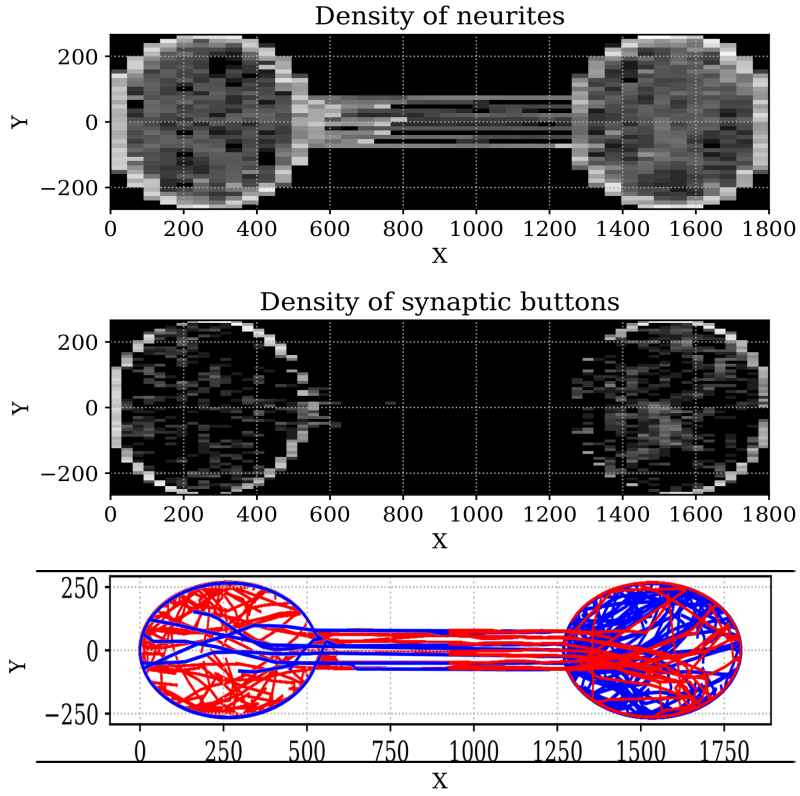


Figure 8.7. NetGrowth generated neurons in diode-like culture The culture implemented by R. Renault in [Renault, 2015] is reproduced with NetGrowth and the relative network is outlined. While the density of neurite is homogeneous in the two chambers the effect of axons invading from left to right is showed by the higher density of synapses in the right chamber: the axons from the left forms several synaptic buttons within the chamber. This configuration is clear in the last picture: left axons in red invade the other chamber with more intensity of right blue axons, this inhomogeneity is due to the shape of the communication channels.

The model and the parameters used are described in the Fig.s together with a sketched Sholl and density analysis. In the second section I have generated two examples of cultures, one as a Petri dish like culture with neurons contained in a circular confinement, the other as the standard diode, borrowed by [Renault, 2015]. For both the culture I sketched a naïve analysis of the network properties, i.e. the adjacency matrix, and graphically described the position of neurite and synapses. After the validation of morphologies will ensure NetGrowth is a effective generator of neuronal morphologies this second phase will be improved with more efficient synapses' creation algorithms.

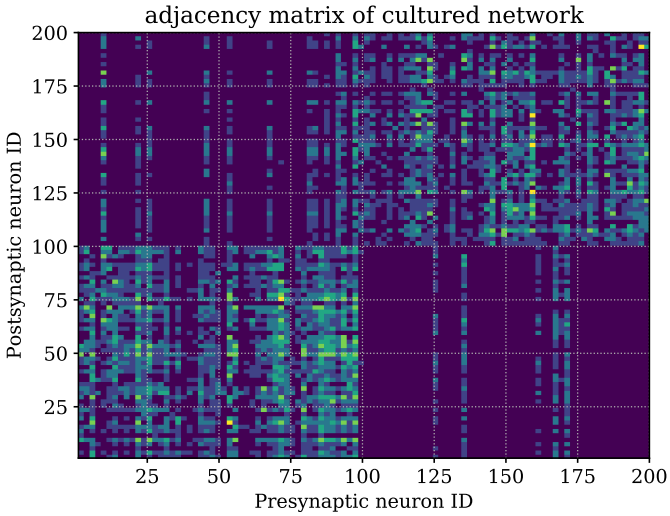


Figure 8.8. NetGrowth generated neurons in diode-like culture, adjacency matrix The adjacency matrix for the diode like culture shows clearly the global properties of the network: some neurons from the left chamber (IDs less then 100), have some neurons with synaptic buttons both in left chamber and in right chamber(IDs more then 100), i.e. the neurons between 75 and 100. Of the neurons in right chamber just a few success to overcome the diode-like structure. This configuration reproduces a perceptron and is one of basic step towards a neuronal based computing device.

Part IV

Conclusions

The NetGrowth simulator has been discussed in the manuscript, focusing on the biophysical models implemented and their relation compared to state of the art models. NetGrowth is a morphology generator for 2D neurons in cultures. It has been designed to produce realistic neuronal morphologies leading to relevant network topologies. It can also be a valid instrument in the quest for relating the microscopic properties of neurons to the macroscopic evidences of cultures and their computational abilities. The introductive work dealt with the positioning of this research project in the vast and quickly evolving field of neuroscience. A huge bibliographic research has been done to evaluate the publications and the models that could be useful for the development of the simulator.

A relevant research project The NetGrowth project is the result of the collaboration among the Neurophysics group at MSC Lab, the group of Catherine Villard at IPGG, Paris, the group of J.Soriano at UVB, Barcelona and E.Moses, Weizmann Institute, Rehovot. These research groups are involved at different levels of neuronal culture studies and they have showed great interest in the realization of the simulator. The existence of a simple IT tool to generate realistic networks, combined to the impressive quality of NEST activity simulator, opens interesting scenarios in the field of neuronal cultures. The costs in terms of time and biological material required to grow a culture are very huge. Hence the opportunity to simulate the experiments reduce considerably the risk of mistakes, NetGrowth can help researchers to choose the relevant shape for the neuronal culture. Although simulations cannot substitute the experimental results and evidences, it can help into predict the outcomes of an experimental setup and offer a comparison tool. The project has revealed interesting also for studies in culture with different substrates. The opportunity offered from the environment interactions are indeed very broad and the realization of mechanical diode, as those proposed in [Renault et al., 2016], is just one of the possible implementations. An experiment that will use the NetGrowth simulator as benchmark will be realized in next months by Soriano and collaborators. The culture will be devised in two or more topographical areas, these areas have different heights and allow upper neurons to descend, while offer confinement to neurons in lower levels. Simulations with NetGrowth will help to devise these areas forecasting the resulting networks.

Devising and implementing models Most of the work I have done on the project is the realization of the simulator infrastructure and the devising and evaluation of the models. The models were presented by increasing complexity: at first the navigation is studied then the mechanical interaction. The second part deals with models of branching and competition. These last models were the more challenging from the analytical point of view: the simplicity they are presented may cover the intense work of modelling in favor of readability of the whole project. Part of the analytical computations are reported in the Appendixes. The navigation part describes the study of random walks and the related generating algorithms: reproducing the navigation has been challenging since a simple random walk was not adequate to reproduce the neurite outgrowth. Usually in physics we deal with random walk that self-cross repeatedly and we look only at the average distance

from the origin. In this case the GC navigation must both reach the target and leave a plausible dendritic tree behind itself. Up to now the neurites intersect each other, we are working on fasciculation and self avoiding model to prevent this inconvenient. The models have been evaluated in this case also for the usability they offer, and for the needs of a clear parameter space for the user. The study on mechanical interactions instead had presented an intense work of algorithmic optimization, switching from a biophysical simple idea to its implementation has revealed tricky with respect to efficiency. The model implemented in the end was efficient and reproduced the experimental evidences. Regarding the branching I had to study and solve non trivial stochastic problems, the first was to derive the interval distribution from a time dependent rate. I studied the problem and approached with classical stochastic instruments without effective success, then I accepted to relax some requirements and obtained an interval distribution that is an effective solution for the needs of the simulator, reproducing correctly the results of the BEST model. Eventually the most of analytical effort regards the ideation, and resolution, of a SDE systems. It was devised to reproduce competition for critical resources. The model was the last of three models which were not analytically solvable. These model are sketched in the thesis but are not deepen since, for the state of the art of NetGrowth, they are not relevant. I have enhanced my ability in approaching stochastic problems and, most of all, in evaluation of the quality of the proposed solutions.

Research perspective The NetGrowth simulator will be soon released for the scientific community. In next future a serious study on the performance will be accomplishes and potential flaws will be solved. Hence the next step is the systematic implementation of methods for synapses generation. Up to now the intersection method is being used, which estimates the numbers of synapses generated by the number of intersections between axon and dendrites, this algorithm can be improved with density field methods.
The implementation of fasciculation among neurites is a required improvement too. On longer terms we expect to develop a simulation environment (SENEC initiative)[1] towards the realization of a growth-activity simulator. We aim to join NetGrowth with the NEST activity simulator through the NNGT library developed by T.Fardet. Either the project misses a sound validation with experiment realised on purpose. For this scope, we plan to ideate and implement experiments with the researchers of IPGG.

Me, student in neuroscience As a theoretical physicist my knowledge in biophysics and neurobiology was very limited. This research project was an opportunity to discover the complexity behind neuronal outgrowth. During the thesis I had the opportunity to meet researchers with very different backgrounds, from people I met in Sissa Neuron Technology Summer School to the researchers of IPGG. Discussing the NetGrowth project with them opened my mind with respect to the multi-faced problems of neuroscience.
I do not think I will keep my way on the biophysics of neuronal growth, but now I

[1] `https://github.com/SENeC-Initiative`

know what is possible and what not in a *in vitro* culture experiment. This experience will help me in collaborating with neurobiologists and experimentalists.

Part V

Appendix

Appendix A

Random Walk characterization

A.0.1 Random Walk, a short hint

Random Walk approach is applied in many fields and the repertoire of problems is pretty huge, we will follow some works on Random Walk in biology [Patlak, 1953], trying to characterize results from *in vitro* experiments and from NetGrowth simulator.
The study of long chain of molecules, as polymers, is a lucky application of Random Walk approach. It's relevant to study their properties depending on the micro physics of the anchorage. Another relevant approach descends from statistical mechanic and shades the light on the macro scale behaviour: a sub-chain of strongly connected elements, can be considered under a coarse grain operation, as an unit, hence the constraints between different units are weaker and the long range behaviour presents other properties than short range. An important objective, among others, is to describe the random walk in a statistical field theory formalism, studying the properties of the polymer chain with the renormalization group. In such formalism the Mean Square Displacement, i.e. the absolute distance between then start and the end of the path, is ω_N, and N is the number of segments.

$$\eta = \lim_{N\to\infty} \frac{\omega_N^2}{N^\gamma} \tag{A.1}$$

is fixed finite and the correct exponent γ is investigated [Procacci et al., 2008]. The algorithms we are implementing are short-range models, since the partial correlation function is an exponential decaying one, and can be demonstrated they ultimately belong to diffusive random walkers as defined in A.1. The literature we are referring to is generally related to polymers. [Cioletti et al., 2014].

A.0.2 Characterization of Random walks: tortuosity, contraction and MSD

To evaluate the morphology produced by the different algorithms we need to search for some statistical properties: Measures offering relevant information of the studied system too, i.e. the distance reached in a certain time by the extruding growth cone. All the statistical measures are computed over the real distance from the soma, measured in micrometers. The assumption made throughout this Appendix is the

process is not stationary: the dynamics is not governed only by relative distances and the origin time has any special meaning. This approximation is relaxed since we expect a strong dependence by time for Growth Cone dynamics.
On the other side we assume the system will relax towards a diffusive Random Walk and the persistence length obtained for the first interval of the neurite is the longest possible during neurite evolution. That is, we expect the GC dynamics will be influenced less by the soma position as far it is from it.

Square distance from the origin

The first element we are evaluating is the distance reached by the Growth Cone. The neuron extrudes its active units, the GC, and they will tread a certain amount of distance before attach to a target dendrite. Understand this is a diffusive or a ballistic behaviour is a crucial point. In this paragraph the length of the segments is considered unitarian and homogeneous throughout the process. Roughly, we can divide the dynamics into two intervals, a super-diffusive (anomalous) and a diffusive one. The former ends when the distance travelled by the growth cone is far larger than the persistence length of the system. In case of real neurons, if presents, the diffusive limit is barely reached, as can be seen in Fig. 6.7. To simulate the growth pattern of real neurons, it's required to know the persistence length of such neurons and reproduce it with the random walker.
In the 2D RW the absolute distance from the initial point is:

$$\langle \Delta X^2 \rangle = \langle (x - x_0)^2 + (y - y_0)^2 \rangle \tag{A.2}$$

the ratio over the curvilinear abscissa can be linear($\alpha = 1$), diffusive ($\alpha = 2$), or present anomalous diffusion with higher powers:

$$\langle \Delta X^2 \rangle \propto t^{\alpha} \tag{A.3}$$

This simple measure is enough to characterize the persistence length of the system. recalling the work of[Tchen, 1989], assuming stationary state, the MSD is:

$$\langle X^2 \rangle = \sum_0^n \langle \vec{b_i} \vec{b_j} \rangle = b^2 \sum_0^n \sum_0^n C(i-j) \tag{A.4}$$

Neglecting the ends the previous equation can be transformed in

$$\langle X^2 \rangle = 2b^2 \sum_{s=0}^{n} \sum_{j=s}^{n} C(s) + b^2 \sum_{j=0}^{n} C(0) \tag{A.5}$$

$$S = \sum_{s=1}^{n} C(s) \tag{A.6}$$

$$M = \sum_{s=1}^{n} sC(s) \tag{A.7}$$

$$\tag{A.8}$$

where S and M can be interpreted as the volume below the correlation curve and its first moment.

Thus, defined $r_0^2 = nb^2$ the diffusive MSD. It's possible the obtain the mean square displacement:

$$\langle \frac{r^2}{r_0^2} \rangle = 1 + 2(S - \frac{M}{n})$$

$$\text{for large n:} \tag{A.9}$$

$$\langle \frac{r^2}{r_0^2} \rangle = 1 + 2S$$

$$\tag{A.10}$$

We have verified this relation for the MCRW in Fig. A.1, The volume S and the moment M are computed numerically.

For an exponentially decaying correlation function states:

$$S = l_p, \quad M = l_p^2$$

The relation between the memory parameter α and the persistence length should be valuated from the partial correlation coefficient $c_k = \langle \vec{b_i} \vec{b_{i+k}} \rangle$, indeed it's possible to demonstrate that for a finite number of partial correlation:

$$\langle r^2/r_0^2 \rangle = \frac{\prod_{j=1}^{k}(1+c_j)}{\prod_{j=1}^{k}(1-c_j)} = 1 + 2l_p \tag{A.11}$$

$$\tag{A.12}$$

But nevertheless we know the partial correlation is an exponentially decaying is still hard to compute it analytically for the MCRW.
Some attempts follow in next paragraph. In next section we will show the relation between the MSD here presented and the statistical evolution of angle variable θ.

Mean Square Displacement of angle variable:

The second relevant measure, linked with previous is the Mean Square Displacement of θ over the distance s.

$$\Delta\theta^2 = (\theta_s - \theta_0)^2 = \theta_s^2$$

In this case it's easy to derive a consistent relation with the persistence length for a short interval. Asserted the global correlation is an exponential dumping function: $C(s) \sim \exp(-l/l_p)$.
For $l \ll l_p$

$$C(s) \sim 1 - \frac{s}{l_p} + \mathcal{O}(l_p^{-2}) \tag{A.13}$$

$$C(s) \sim 1 - \frac{\langle \Delta\theta^2 \rangle}{2} \tag{A.14}$$

$$\langle \Delta\theta^2 \rangle = 2\frac{s}{l_p} \tag{A.15}$$

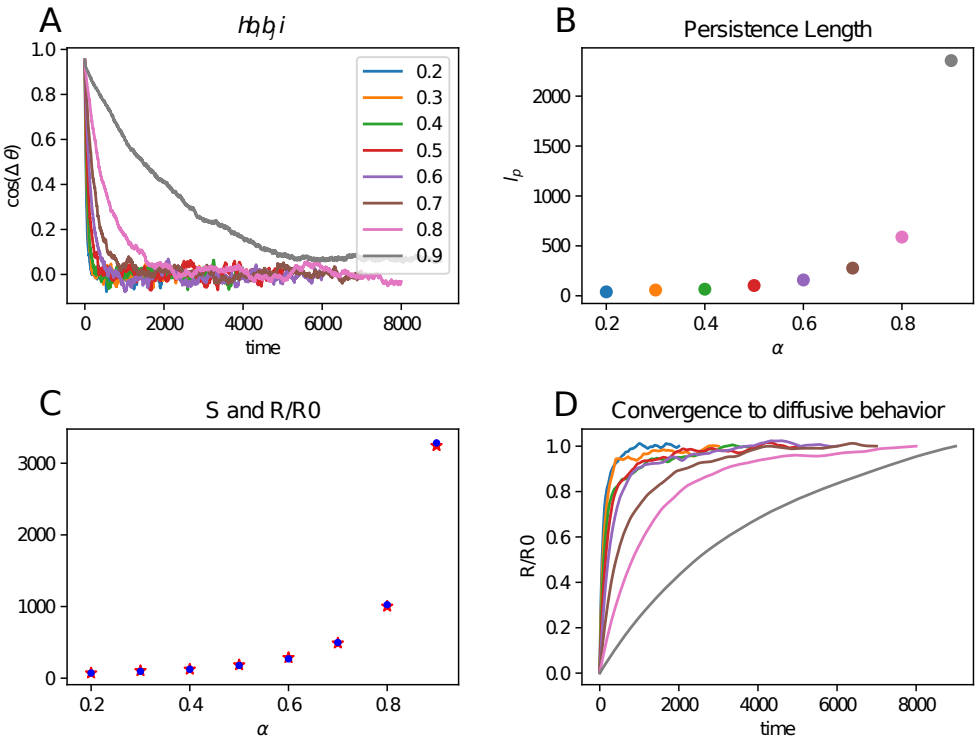


Figure A.1. Preliminary study for exponential decaying memory algorithm
a. The MCRW algorithm, described by Eq. 6.1.3 is executed varying α, then the persistence length is measured as described in 6.4.
b. The persistence length as a function of α and σ, the graphic is fitted by $A * \exp(\alpha * \xi)$ but it shows a super-exponential behaviour. **c**. Eq. The convergence of A.0.2 is verified: red dots correspond to $\frac{R_0^2}{R^2}$, while blue dots are computed through the integral of the correlation curve.
d. The system converges to $\lim_{n\to\infty} R/R0$ faster with lower α. The equilibrium state can be reached in a very long time when α approach to one, and in general reveal the presence of a large transient. In the transient the memory constrained CRW presents a non diffusive behaviour.

This measure naturally defines the *persistence length* as the inverse of the slope: $l_p = m^{-1}_{\langle \Delta\theta^2 \rangle}$. This approach is supported by the article [Martin, 2013] which uses this approximation to estimate the persistence length of microtubules.
To be consistent about persistence length and correlation time let's reformulate it taking into account the variable length of each segment, the previous equation becomes

$$\langle \Delta\theta^2(n) \rangle = 2\frac{s}{l_p b} \tag{A.16}$$

where b is the length of the segment.

Contraction

The contraction is a non standard statistical measure, often encountered in neurons and general biological morphology evaluations. It's defined as the average ratio between the euclidean distance of the growth cone from the soma and the length of the neurite itself; η is always less than 1. It's showed to be an exponential decay for the CRW we are considering and the τ is obtained. The Contraction curve inform us whether the diffusion limit was reached or not, indeed the ratio will change, in exponential manner, until the neurite length is much longer then the persistence length, then will establish on a fixed value which correspond to Eq. A.0.2.

A.0.3 Analytic approach to MCRW

Our interest remains to obtain the persistence length from the parameters in Eq. 6.1.3. Let's make some consideration on this function:

- It loses of physical sense when the angle get larger than π,
- Taylor expansion of Eq. 6.4 requires n remains little.
- To simplify the computation the limit of $\alpha^n \to 0$ would be useful.

To restore limit $\alpha^n \to 0$ a simple consideration can be offered: for large α we don't need to be very close to the origin, since the angular variation will require larger distance to affect the Taylor expansion, while for little α the correlation will reduce to zero in few steps and the limit is valid. Let's look at the average values of the stochastic process, keeping α^n for exactness.

$$\theta_{n+1} = \overline{\theta}_{n+1} + \sigma r_{n+1} \tag{A.17}$$

We note that (A.0.2) can be rewritten:

$$\theta_{n+1} = \frac{1-\alpha}{1-\alpha^{n+1}} \sum_{k=0}^{n} \alpha^{n-k}\theta_k + \sigma r_{n+1} \tag{A.18}$$

Recursive relations For θ:

$$\begin{aligned}\theta_{n+1} &= \frac{1-\alpha}{1-\alpha^{n+1}}\left(\alpha\theta_n + \frac{1-\alpha^n}{1-\alpha}\bar{\theta}_n\right) + \sigma r_{n+1} \\ &= \frac{1-\alpha}{1-\alpha^{n+1}}\left[\alpha\theta_n + \frac{1-\alpha^n}{1-\alpha}\left(\theta_n - \sigma r_n\right)\right] + \sigma r_{n+1} \\ &= \frac{1+\alpha(1-\alpha)-\alpha^n}{1-\alpha^{n+1}}\theta_n + \sigma\left(r_{n+1} - \frac{1-\alpha^n}{1-\alpha^{n+1}}r_n\right)\end{aligned}$$

let

$$z_n = \frac{1+\alpha(1-\alpha)-\alpha^n}{1-\alpha^{n+1}}$$

$$\sigma\left(r_{n+1} - \frac{1-\alpha^n}{1-\alpha^{n+1}}r_n\right) \sim \sigma\left(r_{n+1} - r_n\right)$$

For θ^2:

$$\theta_{n+1}^2 = z^2\theta_n^2 + 2\sigma z\left(r_{n+1} + r_n\right)\theta_n + \sigma^2\left(r_{n+1} + r_n\right)^2$$

Average values If we average, recalling r is a Gaussian with mean zero

$$\begin{aligned}\langle\theta_{n+1}\rangle &= \frac{1+\alpha(1-\alpha)-\alpha^n}{1-\alpha^{n+1}}\langle\theta_n\rangle \\ &\underset{n\to\infty}{\longrightarrow} \qquad\qquad [1+\alpha(1-\alpha)]\langle\theta_n\rangle\end{aligned}$$

this is a simple geometric suite that leads to:

$$\langle\theta_n\rangle = [1+\alpha(1-\alpha)]^n\,\theta_0 \tag{A.19}$$

For the squared value:

$$\begin{aligned}\langle\theta_{n+1}^2\rangle &\sim 1 + z_n^2\langle\theta_n^2\rangle + 2\sigma z_n\langle\left(r_{n+1} - r_n\right)\theta_n\rangle + \sigma^2\langle\left(r_{n+1} - r_n\right)^2\rangle \\ &\sim z_n^2\langle\theta_n^2\rangle + 2\sigma z_n\langle(1-\beta)\sigma r_n^2\rangle + \sigma^2\langle\left(r_{n+1}^2 - 2r_{n+1} + r_n^2\right)\rangle \\ &\sim z_n^2\langle\theta_n^2\rangle + 2\sigma^2(1-\beta)z_n + 2\sigma^2(1-\beta) \\ &\sim z_n^2\langle\theta_n^2\rangle + 2\sigma^2(1-\beta)(z_n+1) \\ &\sim a_n\langle\theta_n^2\rangle + b_n\end{aligned}$$

This equation is exact and can be rewritten starting from θ_0^2:

$$\langle\theta_{n+1}^2\rangle = \prod_i^n a_i\langle\theta_0^2\rangle + \sum_i^n b_i\prod_{k=1}^i a_k \tag{A.20}$$

we should linearize the previous expression for little n and relate to the persistence length through A.16, since the variable n is discrete the simplest way is to evaluate the difference between the MSD in $n=1$ and $n=0$, and setting the initial angle to

zero:

$$\begin{aligned}\frac{d}{dn}\langle\theta^2_{n+1}\rangle|_{n=1} = &\quad (a_1(a_0\langle\theta^2\rangle + b_0) + b_1) - (a_0\langle\theta^2\rangle + b_1)\\ = &\quad b_1 + b_0a_0 - b_0 = b_1 + b_0(a_0 - 1) = b(z_1 + 1) + b(z_0 + 1)(a_0 - 1)\\ = &\quad b(z_1 - z_0 + z_0^3)\\ \frac{1}{l_p} \sim &\quad 2\sigma^2(1-\beta)\left[\frac{1+\alpha-(1-\alpha)^2}{1-\alpha^2} + \left(\frac{1-\alpha(1-\alpha)}{1-\alpha}\right)^3\right]\end{aligned}$$

which is wrong. Indeed the α dependent behaves like an exponent and the persistence length goes to zero for high values of α, which is heuristically and numerically verified incorrect.
The precious computation remains useful to study the long range behaviour of δ^2. It's calculated that for large n there is a arithmetico-geometric suite which can be rewritten as:

$$\langle\theta^2_{n+1}\rangle - \gamma \quad = \quad [1+\alpha(1-\alpha)]^2\left(\langle\theta^2_n\rangle - \gamma\right) \tag{A.21}$$

with

$$\gamma \quad = \quad \frac{b}{1-a} \quad = \quad \frac{2\sigma^2(1-\beta)\left[2+\alpha(1-\alpha)\right]}{\alpha(1-\alpha)\left(2-\alpha(1-\alpha)\right)} \tag{A.22}$$

hence:

$$\langle\theta^2_{n+1}\rangle \quad = \quad [1+\alpha(1-\alpha)]^{2n}\left(\langle\theta^2\rangle_0 - \gamma\right) + \gamma \tag{A.23}$$

This result will be useful in later section when considering the angle distribution on a longer time scale.
Another approach that can be attempted is compute it numerically: *sagemath* [1] a powerful python tool for symbolic computation has been used for it. θ^2 has been computed up to the third term, and then the linearization performed, subtracting the second term. Hoping the function maintains the same rate in the transient.

$$\begin{aligned}\langle\Delta\theta^2\rangle = \langle\theta^2_3\rangle - \langle\theta^2_2\rangle\\ = \left(5a^6 + 10a^5 + 2\left(3a^6 + 9a^5 + 13a^4 + 12a^3 + 7a^2 + 3a + 1\right)\right)\beta^3\\ -2a^3 - 2\left(2\,a^6 + 4\,a^5 + a^4 - 4\,a^3 - 3\,a^2\right)\beta^2\\ -2\,a^2 + 2\,\left(3\,a^5 + 5\,a^4 - 3\,a^3 - 5\,a^2 - 3\,a - 1\right)\beta\\ \frac{\sigma^2}{a^6 + 4\,a^5 + 8\,a^4 + 10\,a^3 + 8\,a^2 + 4\,a + 1}\end{aligned}$$

Neither in this case the function $l_p(\sigma,\beta,\alpha)$ presents the right behaviour and further attempts to perform an analytical estimation of the persistence length won't be done.

Angle Mean Square Displacement and Persistence Length In this section we have tried to measure the persistence length with numerical simulations. To reconstruct the proper function, the one to convert the requested persistence length

[1] www.sagemath.org

into the right adimensional parameters, let's tackle the problem one parameter at time. First l_p is obtained as function of α in a purely memory random walk. This function is illustrated in Fig. A.1 to be super exponential, but since we are interested in moderate values of l_p we can accept the error of considering an exponential function. Then another fit changing σ was attempted, and the following is the formula obtained.

$$l_p = 7\exp(-5\sigma + 12\alpha)\alpha = (\ln(l_p/7) + 5\sigma)/12 \tag{A.24}$$

Despite the expectations the implementation of such formula didn't work, indeed the fit for σ it's to rude and a little change in its value causes a great variation in l_p. The valid alternative was to fit over a 3D manifold, but such computation requires sophisticated approaches, especially if the functional form is unknown. To track values usable in the project have we searched the subset of parameters corresponding to a plausible persistence length, a range between 50 um and 300 um. The table is resumed in a 3D plot in Fig. A.2

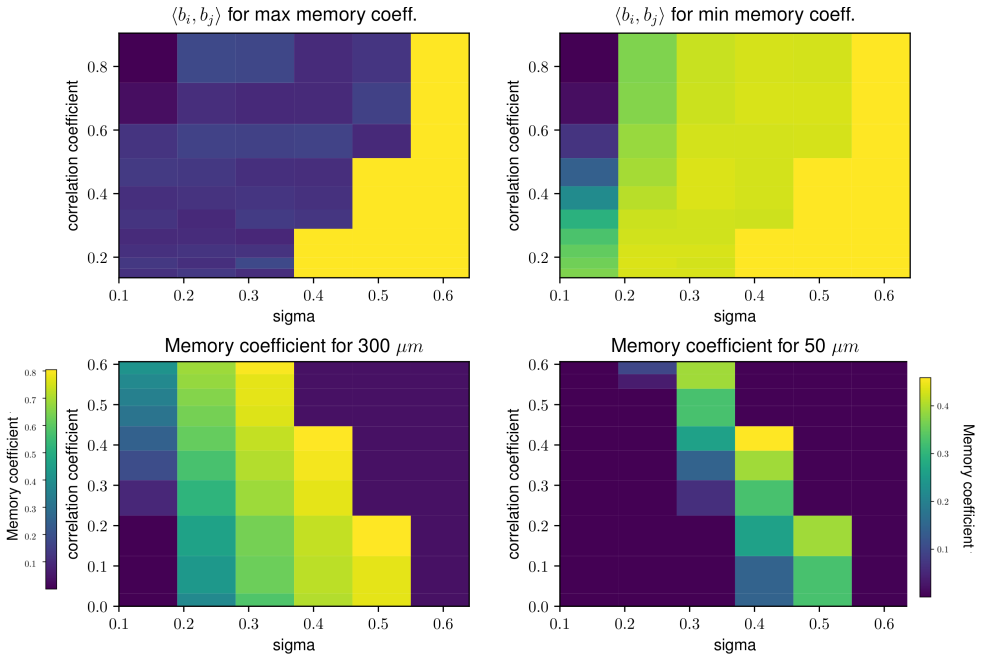


Figure A.2. Plausible values for α, β and σ

Here we present the set of parameters offering a plausible persistence length as result, in the range $l_p \in (50, 300)\mu m$ The grid search was performed in this manner: first the correlated Gaussian parameter β and the angle of view σ were fixed, then the memory parameter α was varied from 0, up to obtain the maximum persistence length. $< b_i b_j >$ was measured through the cosine of $\Delta\theta$.

(**A-B**) show the persistence length reached for each simulation the yellow stripes indicate that was impossible to set a proper α for that set of β, σ. In (**A**) the maximum values of correlation length reached during the grid search. In **B** only a stripe show values about 50 μm: for some parameters even very short memory, $\alpha \sim 0$, presents a persistence length longer than the required interval. (**B-C**) show the max and min values of α for the given persistence length. The graph show the importance of σ in setting the persistence length. The valid values of α are between zero and one, but when the σ is to high (0.6), any values is enough to keep the persistence length in the required range.

Appendix B

Correlated Random Walk: Coloured Noise

The author in [Deserno, 2011] offer a simple algorithm to generate correlated Gaussian, with a correlation coefficient $f = \exp(-1/\tau)$:

$$r_n = f \cdot r_{n-1} + \sqrt{1 - f^2} \cdot \xi_n \tag{B.1}$$

$$r_0 = \xi_0 \tag{B.2}$$

$$C(m - n) = \langle r_n r_m \rangle = f^{m-n} \tag{B.3}$$

In the same paper such system is discussed: Once defined the correlated and uncorrelated random walks as:

$$R_N := \sum_{n=0}^{N} r_n \tag{B.4}$$

$$G_N := \sum_{n=0}^{N} \xi_n \tag{B.5}$$

it's easy to show that:

$$R_N = \sqrt{\frac{1+f}{1-f}} G_N - \frac{f}{1-f} r_N + 1 - \frac{\sqrt{1-f^2}}{1-f} \tag{B.6}$$

$$R_N = \sqrt{\coth \frac{1}{2\tau}} G_N \approx \begin{cases} G_N : \tau \ll 1 \\ \sqrt{2\tau} G_N : \tau \gg 1 \end{cases} \tag{B.7}$$

While the second and third term in Eq.B.6 will be of constant magnitude with time the first term will reflect the behaviour of a classic Random Walk; hence it can be seen that the correlated Gaussian random walk is "faster" than the uncorrelated random walk, even though in both cases the increments are Gaussian deviates with zero mean and unit variance!
More precisely, the mean square displacement of the correlated walk will grow stronger – in the case $\tau \gg 1$ - by a factor of 2τ: Correlation gives distance! This system can be described as Markovian if the space of phase is enlarged to the parameter r_n, and this procedure is thoroughly general.

Appendix C

Fokker Plank

Fokker-Planck

Let's the variable A evolve by the process:

$$\dot{A} = \frac{1}{\tau}(A_M - A) + \xi \tag{C.1}$$

Let $f(A,t)$ be the probability density of A at time t; we have:

$$f(A, t+\Delta t) = \int f(A - A', t) P_{\Delta t}(A'|t, A - A') dA' \tag{C.2}$$

$$= f(A,t) - f\partial_A\langle\Delta A\rangle - \langle\Delta A\rangle\partial_A f + \frac{1}{2} f\partial_A^2\langle(\Delta A)^2\rangle \tag{C.3}$$

$$+\partial_A f \partial_A\langle(\Delta A)^2\rangle + \frac{1}{2}\langle(\Delta A)^2\rangle\partial_A^2 f \tag{C.4}$$

Hence:

$$\partial_t f = -\frac{1}{\tau} f + \left(\frac{A_m}{\tau_A} - \frac{A}{\tau}\right)\partial_A f + \frac{\sigma_\xi^2}{2}\partial_A^2 f = \partial_A\left[\frac{1}{\tau}(A_M - A)f + \frac{\sigma_\xi^2}{2}\partial_A f\right] \tag{C.5}$$

with the boundary condition, since $A \geq 0$,

$$\forall t,\ \partial_A f(0,t) = 0. \tag{C.6}$$

Stationary regime

Once the permanent regime is reached, the Fokker-Plank equation (C.5) becomes:

$$\partial_A\left[\frac{1}{\tau}(A_M - A)f + \frac{\sigma_\xi^2}{2}\partial_A f\right] = 0 \tag{C.7}$$

Hence $\frac{1}{\tau}(A_M - A)f + \frac{\sigma_\xi^2}{2}\partial_A f = 0$ (regularization at $A \to \infty$). Thus $\ln(f) = -\frac{1}{\tau\sigma_\xi^2}(A - A_M)^2 + \text{cst}$, which leads to:

$$f(A) = C\exp\left[-\frac{(A - A_M)^2}{2\sigma_f^2}\right] \tag{C.8}$$

where $\sigma_f = \sigma_\xi \sqrt{\frac{\tau}{2}}$. C is the normalization coefficient given by the integral of the probability distribution to 1 between 0 and ∞:

$$C^{-1} = \int_0^\infty f(A)dA = \int_{-A_M}^\infty \exp\left(-\frac{x^2}{2\sigma_f^2}\right) \tag{C.9}$$

$$= \frac{\sigma_\xi \sqrt{\pi\tau}}{2}\left[1 + \mathrm{erf}\left(\frac{A_M}{\sigma_\xi\sqrt{\tau}}\right)\right] \tag{C.10}$$

Appendix D

Competition, other models

In order to study the competition model we have developed other framework with less success of the implemented one. The model devised are mainly two plus other minor variations Unfortunately I have any graphs or implemented test suite for the models, hence only a theoretical picture will be offered. The first model implemented was a Multiplicative Random Walk, only one equation was governing the amount of received critical resource and the leakage term was absent. The amount of critical resource to be compared with the elongation and retraction thresholds is:

$$da_i = A_0 \frac{a_i \zeta_i}{\sum_j^N a_j \zeta_j} dt + \sqrt{dt} \sigma \xi_i \tag{D.1}$$

This model was too simple and was not able to offer a variegate range of behaviours like we expect from neurite outgrowth. In particular the main failure was in the regulation of CR distribution among growth cones with less resource. Let's perform the derivative received critical resource in respect of the quantity received from another cone in the discrete system, in approximation of Euler discretization, neglecting the stochastic factor

$$\frac{\partial a_i}{\partial a_k} = \frac{\partial}{\partial a_k}(1 + A_0 \frac{a_i \zeta_i}{\sum_j^N a_j \zeta_j} dt) \tag{D.2}$$

$$= A_0 \frac{a_i \zeta_i \zeta_j}{\sum_j^N a_j \zeta_j} + O(\sum_j^N a_j \zeta_j^{-2}) \tag{D.3}$$

Here a reference to Heun http://mathfaculty.fullerton.edu/mathews//n2003/heunsmethod/Heun'sMe
* Neuron, neurite and growth cone properties

Appendix E

Variables of the simulator

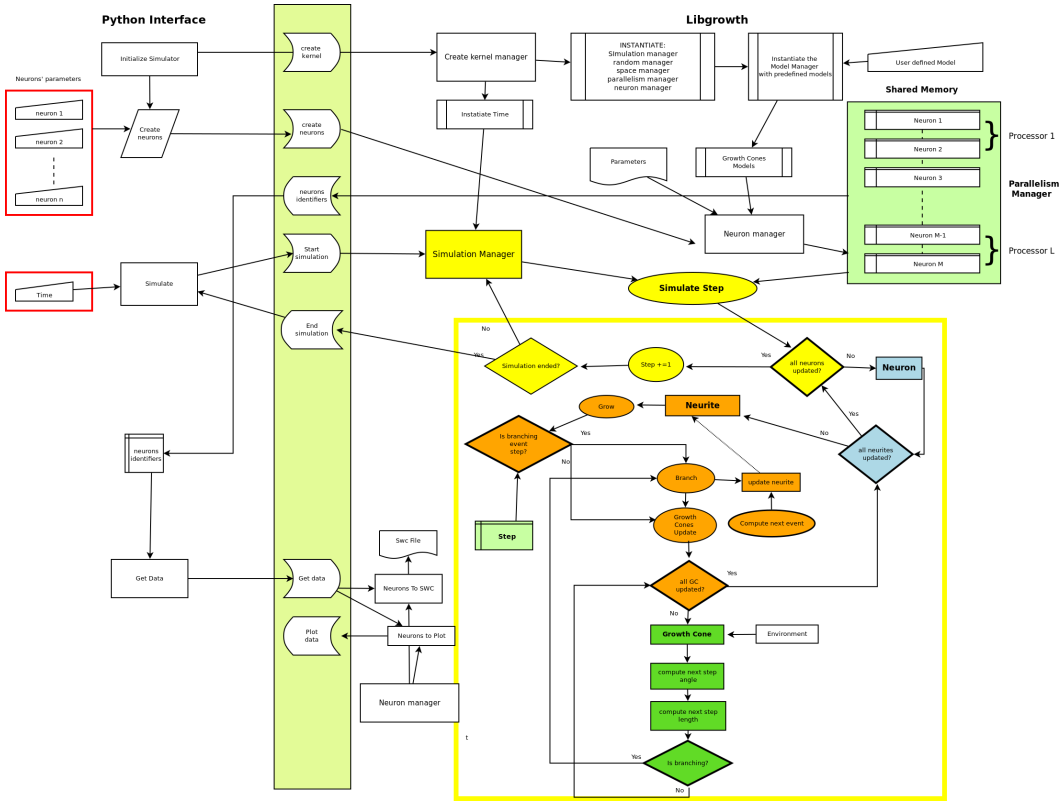
```
//! number of neurites for created neuron [int]
extern const std::string num_neurites;
#defin SOMA_RADIUS // for neurite starts 10 [micrometers]
extern const std::string soma_radius;
#define AXON_DIAMETER 0.6  //[micrometers]
extern const std::string axon_diameter;
#define DENDRITE_DIAMETER 0.6 //[micrometers]
extern const std::string dendrite_diameter;
#define INIT_BRANCHING 0 //initial branching length 0 [micrometers]
extern const std::string initial_branch_length;


/*
 * SPACE SENSING MODEL
 */

extern const std::string duration_retraction;
extern const std::string filopodia_min_number;
extern const std::string filopodia_finger_length;
extern const std::string filopodia_wall_affinity;
extern const std::string max_sensing_angle;
extern const std::string proba_down_move;
extern const std::string proba_retraction;
extern const std::string scale_up_move;
extern const std::string sensing_angle;
extern const std::string speed_ratio_retraction;
extern const std::string substrate_affinity;

#define DURATION_RETRACTION 200.
#define FILOPODIA_MIN_NUM 24
#define FILOPODIA_FINGER_LENGTH 50.
#define FILOPODIA_SUBSTRATE_AFINITY 0.1
#define FILOPODIA_WALL_AFFINITY 2.
#define MAX_SENSING_ANGLE 1.5707963267948966 // 100 degrees max for 1 s
    resol
#define ONE_DEGREE 0.017453292519943295
```

```
#define PROBA_RETRACTION 0.001
#define PROBA_DOWN_MOVE 0.008
#define RW_DELTA_CORR 100.
#define RW_MEMORY_TAU 100.
#define RW_PERSISTENCE_LENGTH 10.
#define SCALE_UP_MOVE 20.
#define SENSING_ANGLE 0.1433
#define SPEED_RATIO_RETRACTION 0.2
#define SPEED_GROWTH_CONE 1.
#define WALL_AFNTY_DECAY_CST 19.098593171027442 // inverse of 3 deg in
    radians


/*
 * CRITICAL MODELS
 */

//! @param tub_topo_coefficient 0.1  [natural]
extern const std::string use_critical_resource;
#define USE_CRITICAL false
extern const std::string CR_use_ratio;
#define CRITICAL_USE_RATIO 1
extern const std::string CR_leakage;
#define CRITICAL_LEAKAGE 6
extern const std::string CR_correlation;
#define CRITICAL_CORRELATION 0.1
extern const std::string CR_variance;
#define CRITICAL_VARIANCE 0.1 //
extern const std::string CR_weight_diameter;
#define CRITICAL_WEIGHT_DIAMETER 1.
extern const std::string CR_weight_centrifugal;
#define CRITICAL_WEIGHT_CENTRIFUGAL 1.

extern const std::string CR_elongation_factor;
#define CRITICAL_ELONGATION_FACTOR 0.5
extern const std::string CR_elongation_th;
#define CRITICAL_ELONGATION_TH 0.35
extern const std::string CR_retraction_factor;
#define CRITICAL_RETRACTION_FACTOR 0.1
extern const std::string CR_retraction_th;
#define CRITICAL_RETRACTION_TH 0.15


extern const std::string CR_neurite_split_th;
#define CRITICAL_SPLIT_TH 250.
extern const std::string CR_neurite_available;
#define CRITICAL_AVAILABLE 100.
extern const std::string CR_neurite_variance;
#define CRITICAL_GEN_VAR 5.
extern const std::string CR_neurite_generated;
#define CRITICAL_GENERATED 150.
extern const std::string CR_neurite_generated_tau;
#define CRITICAL_GEN_TAU 100.
```

```
extern const std::string CR_neurite_delivery_tau;
#define CRITICAL_DEL_TAU 50.
#define CRITICAL_GEN_CORR 0.



/*
 * RANDOM WALK MODEL
 */

extern const std::string random_walk_submodel;
//! @param speed_growth_cone   10     [micormeter/second]
extern const std::string speed_growth_cone;
extern const std::string speed_variance;
//! @param persistenc_length   2000  [micrometer]
extern const std::string rw_persistence_length;
extern const std::string rw_memory_tau;
extern const std::string rw_delta_corr;
//@param sensing angle is choosen from experimental
// data and it's 8.2 degrees

//! RUN AND TUMBLE
extern const std::string rt_persistence_length;
#define RT_PERSISTENCE_LENGTH 100.


//SELF REFERENTIAL MODEL
//
#define SFR_AVOIDANCE_FORCE 1
extern const std::string srf_avoidance_force;
#define SFR_AVOIDANCE_DECAY 2
extern const std::string srf_avoidance_decay;
#define SFR_INERTIAL_FORCE 1
extern const std::string srf_inertial_force ;
#define SFR_INERTIAL_DECAY 2
extern const std::string srf_inertial_decay ;
#define SFR_SOMATROPIC_FORCE 1
extern const std::string srf_somatropic_force;
#define SFR_SOMATROPIC_DECAY 2
extern const std::string srf_somatropic_decay;


/*
 * GROWTH CONE SPLITTING PARAMETERS
 */

extern const std::string gc_split_angle_mean;
extern const std::string gc_split_angle_std;
//! @param van_pelt model for branching probability and direction default:
    True
extern const std::string use_van_pelt;
//! Van_Pelt BEST model parameters
extern const std::string B;
```

```
extern const std::string E;
extern const std::string S;
extern const std::string T;

#define GC_SPLIT_ANGLE_MEAN 98.0 / 180 * 3.14
#define GC_SPLIT_ANGLE_STD 10. / 180 * 3.14
#define USE_VAN_PELT true
#define VP_B 5.
#define VP_E 0.05
#define VP_S 1.
#define VP_T 0.01


/*
 * LATERAL BRANCHING PARAMETERS
 */

extern const std::string use_flpl_branching;
extern const std::string flpl_branching_rate;
extern const std::string use_uniform_branching;
extern const std::string uniform_branching_rate;
extern const std::string lateral_branching_angle_mean;
extern const std::string lateral_branching_angle_std;
extern const std::string diameter_variance;
extern const std::string diameter_eta_exp;

#define ANGLE_IN_DEGREES true
#define DIAMETER_ETA_EXP 1.5
#define DIAMETER_VARIANCE 0.1
#define LATERAL_BRANCHING_ANGLE_MEAN 90 * 3.14 / 180
#define LATERAL_BRANCHING_ANGLE_STD 1. / 180 * 3.14
#define UNIFORM_BRANCHING_RATE 0.001


/*
 * ACTIN WAVE MODEL
 */

//! actin wave trigger or not  False  [bool]
extern const std::string use_actin_waves;
#define USE_ACTIN_WAVES false
//! Actin Waves model parameters
extern const std::string actin_content;
#define ACTIN_CONTENT 0.
extern const std::string actin_content_tau;
#define ACTIN_CONTENT_TAU -1.
extern const std::string actin_wave_speed;
#define ACTIN_WAVE_SPEED 150.
extern const std::string actin_freq;
#define AW_GENERATION_STEP -1.


/*
```

```
 * RECORDERS
 */

extern const std::string event_type;
extern const std::string interval;
extern const std::string level;
extern const std::string observable;
extern const std::string observables;
extern const std::string record_to;
extern const std::string restrict_to;
extern const std::string targets;

extern const signed char lateral_branching;
extern const signed char gc_splitting;
extern const signed char gc_deletion;

extern const std::string num_growth_cones;
```



Appendix F

Neuron morphologies standard format

F.1 Cultured neuron image analysis

I will discuss briefly about image acquisition from experiments and relative analysis. This appendix results by a document I provided to experimental neurobiologist in Septembers, to state our requirements for a effective image analysis and NeetGrowth validation.

F.1.1 Required measures

To test and validate *Netgrowth* models morphological information from experimental observation are necessary.. Since each model concentrate on a different aspect of neuronal morphology development then there are varied possible measures, we can roughly divide into:

Growth cone direction, elongation and guidance To study the mechanical or chemical guidance of the growth cone a detailed information on its path is required, in the growth phase the dynamic of the GC is not expected to be straightly continuous and homogeneous, it's expected to retract, pause and elongate with different interval. A sound comparison between the simulator and the experiment requires information on the dynamics of the GC. Generally these measure can be achieved with fixed camera in bright field, but the quality won't be satisfying; also can be obtained with timelapse of a fluorescent-dye based illumination. NetGrowth can furnish this information with the Recorders, they are still not fully working but we expect we will be able to investigate and compare the dynamics of the Growth process. Generally the dynamics can be stored in TIFF image format.

Neurite branching and topology The shape of neurites requires to be matched with experiment through a graph-like structure. Some simple topological measures (number of internal nodes and leaves) can be performed with a merely topological information, as in Van Pelt and VanOoyen works. Other measures, like Sholl Analysis,

require geometrical information. The TREESToolbox [Cuntz et al., 2012] can help in this.

F.1.2 Cultured neurons images

Formats and Software In order to relate the data generated from a simulator to those acquired through an optical instruments it's required to develop a common language. Such a common language is, computationally speaking, a format. A format is a way to stock data which is standard and uniformly applied by all the contributors. Whichever standard will provide for a finite set of information, this set has to be as large as possible to avoid any loss of details and information.

In neuroscience the process of digitalization has a 20 years backward history. One of the most active researcher in this field is A. Ascoli [1]. He played a main role in the developing of *Neuromorpho.org*[2] the online facility to store and share neuron morphologic data, developed at the George Mason University. The NeuroMorpho (NM) staff works to maintain an updated database of 3D neurons, a research group can upload the result of its research following the application line of the project. The file will be acquired and evaluated and then added to the database. The standard format which NM utilizes is the SWC, an example of stored neurons is Fig. F.1

The standard format for experimentalist is the **TIFF**[3] (Tagged Image File Format) Tiff file can contain time-lapses, they are stored as a series of images on the temporal dimension, like a video. Furthermore it can store different tagged channels, this is particularly relevant when the sample is observed with more than 1 fluorescent dye, e.g. one for tubulin, one for actin, one for calcium, and so on. Tiff format is commonly used by other communities (graphics, physician, etc. . .) but it's a proprietary format. The most common tool to visualize and obtain data from Tiff file is ImageJ, ImageJ allows for channels and timelapses sequences.

Drawbacks

- **Swc** format has no information on time and dynamics, the file is a shoot of the neuron morphology at a precise instant during the growth.
- **Tiff** format has no usable information on neuron topology or geometry, all the information is encoded in pixel.

The NetGrowth team has decided to adhere NeuroMorpho standard format: SWC (Cannon et al., 1998).It is documetned at the end of the appendix. Since the importance this format has in the neuroscience concerning with morphology many tools where developed to translate proprietary format to SWC one: one of them is *NeuronLand*[4] To perform analyses on the neuron we are using *btmorph*[5] tool. It was developed in the framework of another simulator by Torben Nielsen[6]. At present

[1] http://dx.doi.org/10.1016/j.neuron.2013.03.008
[2] http://neuromorpho.org/index.jsp
[3] https://en.wikipedia.org/wiki/TIFF
[4] http://www.neuronland.org/NL.html
[5] http://b-torbennielsen.home.oist.jp/btmorph/index.html
[6] http://b-torbennielsen.home.oist.jp/Software.html

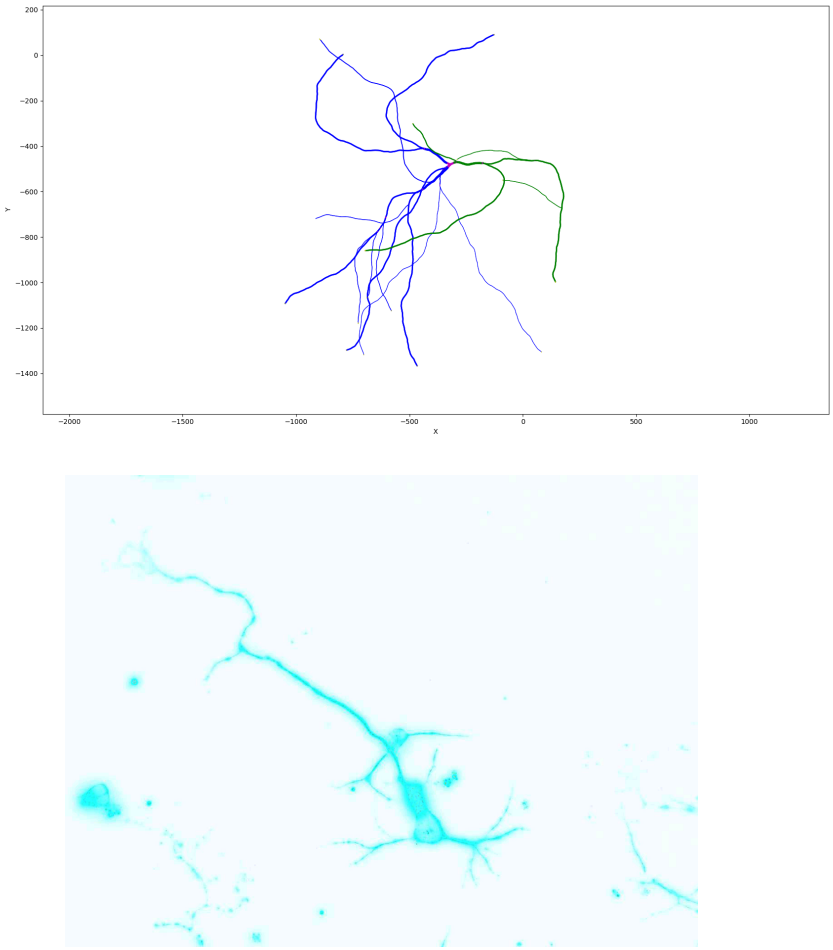


Figure F.1. Dynamics or Morphology *Top*: Neuron image obtained from Swc data from Netgrowth simulato. *bottom* Example of Tiff single time-lapse image, with courtesy of F. Ulloa Severino from Sissa Vincent Torre group

moment the Btmorph github repository is in a messy state since many group were indipendently contributing to it.

We developed a compatibility tool for Netgrowth and we hope keep working on it.

Tiff to Swc converter In order to work together, us(NG team) and whoever want to use our NetGrowth to evaluate is experiments, it is required to have a software to convert TIFF files into SWC. Detect and recognize neurites branching is not an easy task and many tools are available on the Internet, each one with pro and cons. A full reviews of this tool would require a long time, so we concentrated on some tools which are most promising and whose developers are available to interact. Maybe TREESToolbox is the best.

Here it is a list of tools:
`http://www.northeastern.edu/neurogeometry/resources/tools/`
`http://imagej.net/Simple_Neurite_Tracer#Installation`
`https://wiki.colby.edu/display/BI474/ImageJ+Software+-+Neurite+Measurement`
`https://github.com/sgbasel/neuritetracker`
`https://imagescience.org/meijering/publications/download/cyto2004.pdf`
`http://onlinelibrary.wiley.com/doi/10.1002/dneu.20866/full`
`http://fournierlab.mcgill.ca/styled-6/NeuriteTracer.html`

Probabilistic hypotesis density filtering

We had a contact with the developer of this ImageJ plugin[7]. Until now the swc file generated from the program is unusable, mainly for the presence of noise in the file. The following improvement are necessary to use it:

- The program should at least distinguish between different objects. Two neurons should not be in the samefile, or at least they cannot be attached each other in the swc chain. Much more if the attached segments are just noise.
- The program should attach the neurite segments each other and in the proper order. I couldn' check this is done since I cannot plot he SWC with btmorph.
- Should distinguish between neurite and soma, and set the appropriate identifier
- The program should measure the diameter of the neurite and associate each point a dimension, even relative.
- This is the information obtained with phd from same image above.

F.2 SWC format

The three dimensional structure of a neuron can be represented in a SWC format. SWC is a simple Standardized format. A swc neuron is a sequence of cylinders which diameter reflects the neurite diameter in that point. Each elemnt has a parent which

[7] `https://bitbucket.org/miroslavradojevic/phd/overview`

its connected. Every compartment has only one parent and the parent compartment for the first point in each file is always '-1' (if the file does not include the soma information then the originating point of the tree will be connected to a parent of -1). The index for parent compartments are always less than child compartments. Loops and unconnected branches are excluded. All trees should originate from the soma and have parent type 1 if the file includes soma information.

Soma can be a single point or more than one point. When the soma is encoded as one line in the SWC, it is interpreted as a "sphere". When it is encoded by more than 1 line, it could be a set of tapering cylinders (as in some pyramidal cells) or even a 2D projected contour ("circumference"). Each line has 7 fields encoding data for a single neuronal compartment:

`http://www.neuronland.org/NL.html`

```
1. an integer number as compartment identifier\\
2. type of neuronal compartment\\

   * '0'   - undefined
   * '1' - soma
      * '2' - axon
      * '3' - basal dendrite
      * '4' - apical dendrite
      * '5' - fork point
      * '6'- unspecified neurites
      * '7' - glia processes

3. x coordinate of the compartment

4. y coordinate of the compartment

5. z coordinate of the compartment

6. radius of the compartment

7. parent compartment
```

Bibliography

[A Gritsun et al., 2012] A Gritsun, T., Feber, J., and Rutten, W. (2012). Growth dynamics explain the development of spatiotemporal burst activity of young cultured neuronal networks in detail. *Plos Neuroscience*, 7:e43352.

[Amit, 1998] Amit, D. J. (1998). Simulation in neurobiology: theory or experiment? *Trends in Neurosciences*, 21(6):231–237.

[Athamneh et al., 2015a] Athamneh, A. I. M., Cartagena-Rivera, A. X., Raman, A., and Suter, D. M. (2015a). Substrate Deformation Predicts Neuronal Growth Cone Advance. *Biophysical Journal*, 109(7):1358–1371.

[Athamneh et al., 2015b] Athamneh, A. I. M., Cartagena-Rivera, A. X., Raman, A., and Suter, D. M. (2015b). Substrate Deformation Predicts Neuronal Growth Cone Advance. *Biophysical Journal*, 109(7):1358–1371.

[Bak and Fraser, 2003] Bak, M. and Fraser, S. E. (2003). Axon fasciculation and differences in midline kinetics between pioneer and follower axons within commissural fascicles. *Development*, 130(20):4999–5008.

[Baronchelli et al., 2013] Baronchelli, A., Ferrer-i Cancho, R., Pastor-Satorras, R., Chater, N., and Christiansen, M. H. (2013). Networks in Cognitive Science. *Arxiv*, 17(7):348–360.

[Betz et al., 2011] Betz, T., Koch, D., Lu, Y.-B., Franze, K., and Käs, J. A. (2011). Growth cones as soft and weak force generators. *Proceedings of the National Academy of Sciences of the United States of America*, 108(33):13420–5.

[Boffetta Guido, 2012] Boffetta Guido, V. A. (2012). *Probabilità in Fisica Un'introduzione*. Springer.

[Bornschlögl et al., 2013] Bornschlögl, T., Romero, S., Vestergaard, C. L., Joanny, J.-F., Van Nhieu, G. T., and Bassereau, P. (2013). Filopodial retraction force is generated by cortical actin dynamics and controlled by reversible tethering at the tip. *Proceedings of the National Academy of Sciences of the United States of America*, 110(47):18928–33.

[Bosi et al., 2015] Bosi, S., Rauti, R., Laishram, J., Turco, A., Lonardoni, D., Nieus, T., Prato, M., Scaini, D., and Ballerini, L. (2015). From 2D to 3D: novel nanostructured scaffolds to investigate signalling in reconstructed neuronal networks. *Scientific Reports*, 5(1):9562.

[Cheng et al., 2002] Cheng, S., Geddis, M., and Rehder, V. (2002). Local calcium changes regulate the length of growth cone filopodia. *Journal of Neurobiology*, 50:263–75.

[Chua et al., 2014] Chua, J. S., Chng, C. P., Moe, A. A. K., Tann, J. Y., Goh, E. L. K., Chiam, K. H., and Yim, E. K. F. (2014). Extending neurites sense the depth of the underlying topography during neuronal differentiation and contact guidance. *Biomaterials*, 35(27):7750–7761.

[Cioletti et al., 2014] Cioletti, L. M., Dorea, C. C. Y., and Vasconcelos da Silva, S. (2014). Diffusive-Ballistic Transition in Random Polymers with Drift and Repulsive Long-Range Interactions. *Journal of Statistical Physics*, 156(4):760–765.

[Coles and Bradke, 2015] Coles, C. H. and Bradke, F. (2015). Coordinating Neuronal Actin-Microtubule Dynamics. *Current Biology*, 25(15):R677–R691.

[Costa et al., 2010] Costa, L. D. F., Zawadzki, K., Miazaki, M., Viana, M. P., and Taraskin, S. N. (2010). Unveiling the neuromorphological space. *Frontiers in computational neuroscience*, 4(December):150.

[Craig et al., 2012] Craig, E. M., Van Goor, D., Forscher, P., and Mogilner, A. (2012). Membrane tension, myosin force, and actin turnover maintain actin treadmill in the nerve growth cone. *Biophysical Journal*, 102(7):1503–1513.

[Cuntz, 2014] Cuntz, H. (2014). *The Computing Dendrite*, volume 11. Springer.

[Cuntz et al., 2010] Cuntz, H., Forstner, F., Borst, A., and Häusser, M. (2010). One rule to grow them all: A general theory of neuronal branching and its practical application. *PLoS Computational Biology*, 6(8).

[Cuntz et al., 2012] Cuntz, H., Mathy, A., and Häusser, M. (2012). A scaling law derived from optimal dendritic wiring. *Proceedings of the National Academy of Sciences*, 109(27):11014–11018.

[Deserno, 2011] Deserno, M. (2011). How to generate exponentially correlated gaussian random numbers. In *Arxiv*.

[e Parisi, 2010] e Parisi, M. (2010). *Trattatello di probabilità*. Public Domain.

[Fabbro et al., 2012] Fabbro, A., Bosi, S., Ballerini, L., and Prato, M. (2012). Carbon nanotubes: Artificial nanomaterials to engineer single neurons and neuronal networks. *ACS Chemical Neuroscience*, 3(8):611–618. PMID: 22896805.

[Fardet et al., 2018] Fardet, T., Ballandras, M., Bottani, S., Metens, S., and Monceau, P. (2018). Understanding the generation of network bursts by adaptive oscillatory neurons. *In processing*, 12.

[Franze and Guck, 2010] Franze, K. and Guck, J. (2010). The biophysics of neuronal growth. *Reports on Progress in Physics*, 73(9):094601.

[Gallo, 2011] Gallo, G. (2011). The cytoskeletal and signaling mechanisms of axon collateral branching. *Developmental Neurobiology*, 71(3):201–220.

[Gangaraju and Lin, 2009] Gangaraju, V. K. and Lin, H. (2009). MicroRNAs: key regulators of stem cells. *Nature Reviews Molecular Cell Biology*, 10(2):116–125.

[Goldberg, 2003] Goldberg, J. L. (2003). How does an axon grow ? *Journal of Neurobiology*, (650):941–958.

[Grienberger and Konnerth, 2012] Grienberger, C. and Konnerth, A. (2012). Imaging Calcium in Neurons. *Neuron*, 73(5):862–885.

[Hely, 2001] Hely, T. G. B. (2001). A Computational Model of Dendrite Elongation and Branching Based on MAP2 Phosphorylation. *Journal of Theoretical Biology*, 210(3):375–384.

[Hjorth et al., 2014] Hjorth, J. J. J., Van Pelt, J., Mansvelder, H. D., and Van Ooyen, A. (2014). Competitive dynamics during resource-driven neurite outgrowth. *PLoS ONE*, 9(2).

[Jülicher et al., 2007] Jülicher, F., Kruse, K., Prost, J., and Joanny, J. F. (2007). Active behavior of the Cytoskeleton. *Physics Reports*, 449(1-3):3–28.

[Kahn and Baas, 2016] Kahn, O. I. and Baas, P. W. (2016). Microtubules and Growth Cones: Motors Drive the Turn. *Trends in Neurosciences*, 39(7):433–440.

[Kiddie et al., 2004] Kiddie, G., McLean, D., Van Ooyen, A., and Graham, B. (2004). Biologically plausible models of neurite outgrowth. *Progress in Brain Research*, 147(SPEC. ISS.):67–80.

[Koene et al., 2009] Koene, R. A., Tijms, B., Van Hees, P., Postma, F., De Ridder, A., Ramakers, G. J. A., Van Pelt, J., and Van Ooyen, A. (2009). NETMORPH: A framework for the stochastic generation of large scale neuronal networks with realistic neuron morphologies. *Neuroinformatics*, 7(3):195–210.

[Lindauer, 1996] Lindauer, M. (1996). *Message without words, How animals communicate*, volume 214. Mondadori.

[London and Häusser, 2005] London, M. and Häusser, M. (2005). Dendritic computation. *Annal Review, Neuroscience*.

[Lowery and Vactor, 2009] Lowery, L. A. and Vactor, D. V. (2009). The trip of the tip: understanding the growth cone machinery. *Nature Reviews Molecular Cell Biology*, 10(5):332–343.

[Luczak, 2006] Luczak, A. (2006). Spatial embedding of neuronal trees modeled by diffusive growth. *Journal of Neuroscience Methods*, 157(1):132–141.

[Martin, 2013] Martin, D. S. (2013). *Measuring microtubule persistence length using a microtubule gliding assay*, volume 115. © 2013 Elsevier, Inc. All rights reserved., 2 edition.

[Massimini, 2013] Massimini, T. (2013). Nulla di più grande. *Baldini & Castoldi (I saggi)*.

[Memelli et al., 2013] Memelli, H., Torben-Nielsen, B., and Kozloski, J. (2013). Self-referential forces are sufficient to explain different dendritic morphologies. *Frontiers in neuroinformatics*, 7:1.

[Metzler and Klafter, 2004] Metzler, R. and Klafter, J. (2004). The restaurant at the end of the random walk: recent developments in the description of anomalous transport by fractional dynamics. *Journal of Physics A: Mathematical and General*, 37(31):R161–R208.

[Mogilner and Rubinstein, 2005] Mogilner, A. and Rubinstein, B. (2005). The physics of filopodial protrusion. *Biophysical journal*, 89(2):782–95.

[Ooyen, 2004] Ooyen, A. V. (2004). Competition in neurite outgrowth and the development of nerve connections. *Progress in Brain Research*, 147(SPEC. ISS.):81–99.

[Orlandi et al., 2013a] Orlandi, J. G., Soriano, J., Alvarez-lacalle, E., Teller, S., and Casademunt, J. (2013a). Noise focusing : the emergence of coherent activity in neuronal cultures supplementary information. *Nature Physics*.

[Orlandi et al., 2013b] Orlandi, J. G., Soriano, J., Alvarez-lacalle, E., Teller, S., and Casademunt, J. (2013b). Noise focusing : the emergence of coherent activity in neuronal cultures supplementary information. *Nature Physics*.

[O'Toole et al., 2008] O'Toole, M., Lamoureux, P., and Miller, K. E. (2008). A physical model of axonal elongation: force, viscosity, and adhesions govern the mode of outgrowth. *Biophysical journal*, 94(7):2610–20.

[Oşan et al., 2011] Oşan, R., Su, E., and Shinbrot, T. (2011). The interplay between branching and pruning on neuronal target search during developmental growth: Functional role and implications. *PLoS ONE*, 6(10).

[Patlak, 1953] Patlak, C. S. (1953). Random walk with persistence and external bias. *The bulletin of mathematical biophysics*, 15(3):311–338.

[Polavaram et al., 2014] Polavaram, S., Gillette, T. A., Parekh, R., and Ascoli, G. A. (2014). Statistical analysis and data mining of digital reconstructions of dendritic morphologies. *Front Neuroanat*, 8(December):138.

[Procacci et al., 2008] Procacci, A., Sanchis, R., and Scoppola, B. (2008). Diffusive-ballistic transition in random walks with long-range self-repulsion. *Letters in Mathematical Physics*, 83(2):181–187.

[Rauch et al., 2013] Rauch, P., Heine, P., Goettgens, B., and Käs, J. A. (2013). Forces from the rear: Deformed microtubules in neuronal growth cones influence retrograde flow and advancement. *New Journal of Physics*, 15(July 2017).

[Recho et al., 2016] Recho, P., Jerusalem, A., and Goriely, A. (2016). Growth, collapse, and stalling in a mechanical model for neurite motility. *Physical Review E - Statistical, Nonlinear, and Soft Matter Physics*, 93(3).

[Renault, 2015] Renault, R. (2015). Emergent design of and neuronal devices. In *Thèse de doctorat*.

[Renault et al., 2016] Renault, R., Durand, J.-B., Viovy, J.-L., and Villard, C. (2016). Asymmetric axonal edge guidance: a new paradigm for building oriented neuronal networks. *Lab on a chip*, 16:2188–91.

[Roth et al., 2014] Roth, S., Bisbal, M., Brocard, J., Bugnicourt, G., Saoudi, Y., Andrieux, A., Gory-Fauré, S., and Villard, C. (2014). Shaping neurons: how morphological constraints affect axonal polarity. *Neuroscience*.

[Segev et al., 2017] Segev, R., Ben-jacob, E., and Faculty, B. S. (2017). Generic modeling of chemotactic based self-wiring of neural networks. *Arxiv*.

[Shannon, 1948] Shannon, C. E. (1948). A mathematical theory of communication. *The Bell System Technical Journal*, 27(July 1928):379–423.

[Shefi et al., 2004] Shefi, O., Harel, A., B. Chklovskii, D., Ben-Jacob, E., and Ayali, A. (2004). Biophysical constraints on neuronal branching. *Neurocomputing*, 58-60:487–495.

[Sholl, 1953] Sholl, D. (1953). Dendritic organization in the neurons of the visual and motor cortices of the cat. *Journal of anatomy*, 87(4):387—406.

[Singh and Miller, 2005] Singh, K. K. and Miller, F. D. (2005). Activity regulates positive and negative neurotrophin-derived signals to determine axon competition. *Neuron*, 45(6):837–845.

[Smirnov, 2014] Smirnov, M., a. (2014). The effects of confinement on neuronal growth cone morphology and velocity. *Biomaterials 35, 6750–6757*.

[Snider et al., 2010] Snider, J., Pillai, A., and Stevens, C. F. (2010). A Universal Property of Axonal and Dendritic Arbors. *Neuron*, 66(1):45–56.

[Sokolov et al., 2017] Sokolov, A. A., Miall, R. C., and Ivry, R. B. (2017). The Cerebellum: Adaptive Prediction for Movement and Cognition. *Trends in Cognitive Sciences*, 21(5):313–332.

[Suter and Miller, 2011] Suter, D. M. and Miller, K. E. (2011). The emerging role of forces in axonal elongation. *Progress in Neurobiology*, 94(2):91–101.

[Szebenyi et al., 1998] Szebenyi, G., Callaway, J. L., Dent, E. W., and Kalil, K. (1998). Interstitial branches develop from active regions of the axon demarcated by the primary growth cone during pausing behaviors. *The Journal of neuroscience : the official journal of the Society for Neuroscience*, 18(19):7930–7940.

[Tchen, 1989] Tchen, C. M. (1989). Random flight with multiple partial correlations. *Arxiv*, 214(1952).

[Torben-Nielsen and De Schutter, 2014] Torben-Nielsen, B. and De Schutter, E. (2014). Context-aware modeling of neuronal morphologies. *Frontiers in neuroanatomy*, 8(September):92.

[Tsigankov and Koulakov, 2009] Tsigankov, D. and Koulakov, A. (2009). Optimal axonal and dendritic branching strategies during the development of neural circuitry. *Frontiers in neural circuits*, 3(November):18.

[van Ooyen et al., 2014] van Ooyen, A., Carnell, A., de Ridder, S., Tarigan, B., Mansvelder, H. D., Bijma, F., de Gunst, M., and van Pelt, J. (2014). Independently outgrowing neurons and geometry-based synapse formation produce networks with realistic synaptic connectivity. *PloS one*, 9(1):e85858.

[Van Pelt et al., 1997] Van Pelt, J., Dityatev, A. E., and Uylings, H. B. M. (1997). Natural variability in the number of dendritic segments: Model-based inferences about branching during neurite outgrowth. *Journal of Comparative Neurology*, 387(3):325–340.

[van Pelt et al., 2003a] van Pelt, J., Graham, B. P., Uylings, H. B. M., and Pelt, J. V. (2003a). Formation of dendritic branching patterns. *Modeling Neural Development*, (January 2003):75–94.

[van Pelt et al., 2003b] van Pelt, J., Graham, B. P., Uylings, H. B. M., and Pelt, J. V. (2003b). Formation of dendritic branching patterns. *Modeling Neural Development*, (January 2003):75–94.

[Van Pelt and Uylings, 1999] Van Pelt, J. and Uylings, H. B. M. (1999). Modeling the natural variability in the shape of dendritic trees: Application to basal dendrites of small rat cortical layer 5 pyramidal neurons. *Neurocomputing*, 26-27:305–311.

[Van Pelt and Uylings, 2003] Van Pelt, J. and Uylings, H. B. M. (2003). Growth functions in dendritic outgrowth. *Brain and Mind*, 4(1):51–65.

[Velte and Masland., 1999] Velte, T. J. and Masland., R. H. (1999). Action potentials in the dendrites of retinal ganglion cells. *Journal of Neurophysiology, 81(3) :1412–1417*.

[Von Der Ecken et al., 2015] Von Der Ecken, J., Müller, M., Lehman, W., Manstein, D. J., Penczek, P. A., and Raunser, S. (2015). Structure of the F-actin-tropomyosin complex. *Nature*, 519(7541):114–117.

[Wen and Zheng, 2006] Wen, Z. and Zheng, J. Q. (2006). Directional guidance of nerve growth cones. *Current Opinion in Neurobiology*, 16(1):52–58.